**FIELD MANUAL SERIES**

# Modern Process Improvement with Python

## White Belt v0.1.0

Shane M. Wilkins

October 1, 2026

# Table of contents

**Home** **1**
License . . . . . 1
Changelog . . . . . 1

**A Note on AI Assistance** **3**
How I Work with These Tools . . . . . 3
What I Have Found Useful—and What Needs Scrutiny . . . . . 4
Keeping Hold of the Work . . . . . 4

**I. Introduction** **6**

**1. A First Taste** **7**
1.1. What Is This Book About? . . . . . 7
1.2. What Is Lean Six Sigma? . . . . . 12
1.3. Who Is This Book For? . . . . . 14
1.4. Why Another Lean Six Sigma Book? . . . . . 15
1.5. Beyond Traditional Lean Six Sigma . . . . . 16
1.6. Conclusion . . . . . 19

**2. The Method and the Madness** **20**
2.1. A Curriculum Built Around Capability . . . . . 20
2.2. The Three Coordinates of the Curriculum . . . . . 24
2.3. Putting It Together: The Learning Map . . . . . 29
2.4. From Learning Objectives to Practice . . . . . 30
2.5. Anatomy of a Belt and Its Chapters . . . . . 32
2.6. How to Use This Book . . . . . 34
2.7. Conclusion . . . . . 35

**II. White Belt** **36**

**3. Threshold Story: The Friday Pickup Problem** **37**
3.1. Learner Starting Point . . . . . 38
3.2. Belt Promise . . . . . 38
3.3. Why This Case? Why Now? . . . . . 38
3.4. What You Will Learn in This Belt . . . . . 38

3.5. Required Equipment . . . 39
3.6. White Belt Artifact Roadmap . . . 39
3.7. Capstone Preview . . . 40

**4. Lean Foundations 41**
4.1. Why This Chapter Matters . . . 41
4.2. Learning Objectives . . . 41
4.3. Foundational Concepts . . . 42
4.4. Process Runs and Process Variants . . . 47
4.5. Two Pillars . . . 48
4.6. Conclusion . . . 51
4.7. Exercises . . . 52
4.8. Chapter Summary . . . 53

**5. The Five Principles of Lean 54**
5.1. Why This Chapter Matters . . . 54
5.2. Learning Objectives . . . 54
5.3. Five Principles Overview . . . 55
5.4. Defining Value from the Customer Point of View . . . 56
5.5. Mapping the Process . . . 57
5.6. Creating Flow . . . 58
5.7. Establishing Pull . . . 61
5.8. Pursue Perfection . . . 64
5.9. Five Principles, Two Pillars . . . 64
5.10. Conclusion . . . 66
5.11. Exercises . . . 66
5.12. Chapter Summary . . . 67

**6. A Structured Approach to Process Improvement 69**
6.1. Why This Chapter Matters . . . 69
6.2. Learning Objectives . . . 69
6.3. The PDCA Cycle . . . 70
6.4. Wash n' Fold Project: Plan . . . 72
6.5. Conclusion . . . 75
6.6. Exercises . . . 75
6.7. Chapter Summary . . . 78

**7. Following the Work: Building the Event Log 79**
7.1. Why This Chapter Matters . . . 79
7.2. Learning Objectives . . . 79
7.3. Introduction to Process Mapping . . . 80
7.4. Wash n' Fold Project: Follow the Orders . . . 82
7.5. Defining the Process Boundaries . . . 83
7.6. Prepare the Event Log . . . 85
7.7. Follow Work Units Through Major Activities and Variants . . . 87
7.8. Record Who Owns Each Step . . . 89
7.9. Record Timestamps, Queues, Rework, and Interruptions . . . 90

7.10. Exercises . . . 92
7.11. Chapter Summary . . . 95
7.12. Further Reading . . . 95

**8. Building the Current-State Process Map 96**
8.1. Why This Chapter Matters . . . 96
8.2. Learning Objectives . . . 96
8.3. Wash n' Fold Project: From Records to a Map . . . 97
8.4. Calculate Each Activity Record's P/T, W/T, and S/T . . . 97
8.5. Build the Map Summary Sheet . . . 101
8.6. Draft the Visual Map (SPPM Conventions) . . . 107
8.7. Select Queues and Draw the Current-State Map . . . 108
8.8. Verify the Map with a Participant . . . 110
8.9. Exercises . . . 112
8.10. Chapter Summary . . . 115

**9. Measuring and Interpreting Process Performance 116**
9.1. Why This Chapter Matters . . . 116
9.2. Learning Objectives . . . 117
9.3. From Events to Customer Observations . . . 118
9.4. What Are We Actually Summarizing? . . . 119
9.5. From Activity Records to Complete Orders . . . 123
9.6. Typical, Tail, and Promise Performance . . . 126
9.7. Read the Process as a System . . . 131
9.8. Takt Time, Demand, and Capacity . . . 131
9.9. Process Cycle Efficiency . . . 132
9.10. WIP, Throughput, and Little's Law . . . 134
9.11. Build the Baseline Performance Readout . . . 136
9.12. Exercises . . . 138
9.13. Chapter Summary . . . 141

**10. The Eight Wastes 142**
10.1. Why This Chapter Matters . . . 142
10.2. Learning Objectives . . . 142
10.3. Wash n' Fold Project: Do . . . 143
10.4. Exercises . . . 152
10.5. Chapter Summary . . . 154

**11. Designing the Improved Workplace: Flow, Pull, and 5S 155**
11.1. Why This Chapter Matters . . . 155
11.2. Learning Objectives . . . 155
11.3. Wash n' Fold Project: Do (Continued) . . . 156
11.4. From Waste Diagnosis to Flow and Pull . . . 156
11.5. The 5S Elements . . . 161
11.6. When 5S Is Not Enough . . . 171
11.7. Exercises . . . 173
11.8. Chapter Summary . . . 175

**12. Check and Act: What Did We Actually Improve? 176**
12.1. Why This Chapter Matters . . . 176
12.2. Learning Objectives . . . 176
12.3. Wash n' Fold Project: Check . . . 177
12.4. Wash n' Fold Project: Act . . . 181
12.5. Exercises . . . 184
12.6. Chapter Summary . . . 185

**13. White Belt Capstone: Run the Cycle 186**
13.1. Why This Chapter Matters . . . 186
13.2. Learning Objectives . . . 186
13.3. Capstone Assignment . . . 187
13.4. Capstone Crosswalk . . . 191
13.5. What Counts as a Defensible Result . . . 192
13.6. Exercises . . . 192
13.7. Chapter Summary . . . 194
13.8. Further Reading . . . 194

**III. Readings 196**

**14. The Student, the Fish and Agassiz 197**

**IV. Appendices 202**

**15. Lean Six Sigma Learning Map 203**
15.1. How to Read the Map . . . 203
15.2. Belt Levels . . . 204
15.3. Knowledge Domains . . . 204
15.4. Domain-by-Belt Crosswalk . . . 205
15.5. Auditing the Curriculum Against the Map . . . 216

**16. The First Reckoning: What Did You Make One? 217**
16.1. Before You Begin . . . 217
16.2. Act I: Six Circles of Stone . . . 218
16.3. Act II: The Count Becomes an Account . . . 228
16.4. Act III: Enough for Whom? . . . 238
16.5. Act IV: Re'u's Mark . . . 249
16.6. Back to the Stones . . . 257
16.7. Coda: The Beginning . . . 259
16.8. Arithmetic Review and Practice . . . 261
16.9. Chapter Summary . . . 286
16.10. Historical Note and Further Reading . . . 286

**V. Exercise Answers and Guidance** **289**

**17. Solutions to White Belt Exercises** **290**
17.1. Lean Foundations . . . . . 290
17.2. The Five Principles of Lean . . . . . 293
17.3. A Structured Approach to Process Improvement . . . . . 296
17.4. Following the Work: Building the Event Log . . . . . 302
17.5. Building the Current-State Process Map . . . . . 306
17.6. Measuring and Interpreting Process Performance . . . . . 312
17.7. The Eight Wastes . . . . . 318
17.8. Designing the Improved Workplace: Flow, Pull, and 5S . . 321
17.9. Check and Act: What Did We Actually Improve? . . . . . 327
17.10. White Belt Capstone: Run the Cycle . . . . . 331

**VI. Reference** **333**

**18. Glossary** **334**

**19. Bibliography** **350**

# Home

This book provides a thorough overview of Lean Six Sigma process improvement methodology. It expands the traditional process improvement curriculum with new computational tools designed with the practitioner in mind.

The current release provides the introduction and complete White Belt learning path, with an arithmetic appendix, practice data, worksheets, and worked examples. Later belts are still under development. Start with A First Taste, then use How This Book Is Organized to plan your study. White Belt begins with observation, elementary arithmetic, and pencil-and-paper work; no programming experience is required.

Read A Note on AI Assistance for how AI tools contributed to the text and accompanying resources, and the author's responsibility for the published work.

#### License

Unless otherwise noted, the original book text and curriculum content in this project are licensed under CC BY 4.0, and the original source code is licensed under the MIT License.

Some images, readings, quotations, datasets, and other third-party materials may be covered by their own licenses or permissions and are not relicensed by this notice. See the repository's LICENSE.md file and any local credit notes for details.

#### Changelog

This White Belt edition adds a connected observation-to-improvement workflow with matching blank worksheets and completed Wash n' Fold examples. The downloads include short practice records, a printable process map, participant verification, baseline readouts, and one-page standard work.

The arithmetic appendix explains the foundational calculations through Nisaba's story and follow-up practice. Public chapter and capstone exercises have answers or acceptance guidance in the HTML edition and PDF solutions appendix; operational assessments are available separately through the LMS.

# A Note on AI Assistance

I have made extensive use of artificial intelligence tools in writing and revising this book and developing the accompanying software, visualizations, and simulations. That assistance includes suggesting and revising prose, implementing and debugging code, preparing figures and worksheets, checking consistency, and reviewing the alignment of exercises with learning objectives. It has shaped the work you are reading, so it deserves a direct acknowledgment. The responsibility for what I publish remains mine, including its errors.

### How I Work with These Tools

My practice is generally to treat an AI assistant as a capable, fast-working junior colleague who needs direction and whose work needs review. I begin with notes about what I am trying to accomplish, the steps required, and the difficulties I expect to encounter. For prose, I usually develop a first draft before asking the assistant to read it alongside those notes. I ask it to identify problems in reasoning, pacing, structure, terminology, and expression, then discuss the findings before authorizing revisions. For software and learner resources, I also use it to implement changes under my direction and help build checks of the resulting behavior.

I keep a separate guide to the voice I am trying to achieve and ask the assistant to use it when revising prose. A technically acceptable sentence can still flatten a joke, blur a distinction, or replace an argument with something that merely sounds respectable. Those are losses, even when the new sentence is smoother.

I also invite the assistant to ask clarifying questions. This has been one of the most useful parts of the process: sometimes a question reveals a distinction or dependency I had not considered. After revisions, I try to put the work aside and return with fresh eyes. There are usually more changes to make: sharpening an explanation, adjusting a metaphor, or putting a joke back where it belongs. AI assistance has shortened the distance between a

reasonable first draft and a finished piece of work; it has not removed the need to travel that distance.

### What I Have Found Useful—and What Needs Scrutiny

In my experience, these tools are particularly useful for highly structured tasks: repetitive formatting, boilerplate code, a visualization with clearly specified data, or a consistency check across several documents. They can also be useful partners in diagnosing bugs, suggesting revisions, and identifying gaps in an explanation. The result depends heavily on the instructions, the context, and the particular task. These are observations about my working practice, not a permanent ranking of what every AI system can do.

The difficulties become more serious when the task requires sustained judgment across a chapter, a curriculum, or a software system. An assistant can produce a plausible explanation that misses the point, a polished paragraph that no longer says what I meant, or a calculation that quietly uses the wrong boundary. A proposed source or quotation still needs checking against the source itself. Code that runs still needs to perform the intended operation. A favorable AI review is not independent evidence that a learner can use the material successfully.

The same distinction matters for the examples in this book. AI assistance in preparing a case, calculation, or simulation does not turn it into an observed business result. The fictional Wash n' Fold case remains instructional fiction; its records and modeled changes must be read within the assumptions stated in the chapters.

### Keeping Hold of the Work

AI is a little like improv. The improviser's motto is "Yes, and . . . " which is brilliant for keeping a scene moving because the performer accepts what has been offered and elaborates upon it. In the same way, an AI assistant can be very good at keeping movement going, including when the premise should have been questioned several paragraphs ago. That can be a fun way to explore an idea. It can also produce a horrible tangled mess of nonsense if nobody stops to ask what the work is supposed to establish.

I think these tools can be fantastically useful when you know enough about the subject to direct the work and examine the result critically. If you ask them to substitute for knowledge you do not have, their fluency can make the problem harder to notice. The more the pieces need to hang together—as a section, a chapter, or a working system—the more carefully the author has

to specify and inspect their connections. That is the discipline I have tried to follow here.

**Part I.**

# Introduction

# 1.

# A First Taste

> Fine things are difficult. — Plato, *Republic* 435c

### 1.1. What Is This Book About?

A *business process* is a set of steps designed to produce a product or provide a service. Most of our work life happens in performing such processes. On a daily basis, you probably encounter many different processes, even if you don't recognize them as such. There were many processes involved in getting water and electricity to your home, for instance. You don't have to know about those processes; you just turn the tap and the water's there. A trip to the mailbox exposes a more legible process: Somebody wrote a letter, addressed an envelope, the postal service routed and delivered it, and now it's in your box.

From a surgeon operating in a hospital, to a bookkeeper logging transactions in a ledger, to a teacher building a lesson plan, everywhere we look, we find work being accomplished through a repeatable set of steps. In other words, we find processes.

We also encounter broken processes, whose failures manifest as delays, missing or incorrect orders, unexpectedly high costs, or botched haircuts. If you reflect for a few moments, you can probably identify some familiar processes that need improving. But how do we make badly behaved processes better?

Try an *hors d'oeuvre* to whet your appetite.

> You go to the Department of Motor Vehicles to renew your driver's license. You receive a form to fill out that asks you to validate your current address. You take a ticket and wait. When your number is called you return the form and wait to have your picture taken.

> Then the person who takes the pictures goes to lunch, so you wait again. Finally, the staffer returns, you get your picture taken, and you go home where you can wait a final time to receive your new license in the mail.

Overall, it's a pretty frustrating experience for you as the customer. A common and understandably human reaction to this frustration is to blame somebody, likely whichever staff member happens to be most conspicuous at the moment. That's not process improvement.

Process improvement asks us to take a step back, to see the whole process as a system that accepts inputs in the form of material and information, and uses human labor and machinery to transform them into outputs. Process improvement studies the behavior of such systems in order to engineer better systems with better behavior.

Let's illustrate by supposing the Department of Motor Vehicles has asked us to form a small team to help improve this process. For the sake of the illustration, let's stipulate two things. First, the experience above is not an anomaly: significant numbers of customers are unhappy with the wait time. Let's say customer feedback indicates customers are satisfied with their experience at the DMV only if their "total visit time" from walking in the door to leaving with their business successfully completed is less than 30 minutes. On the busiest days, when 50 customers arrive for license renewals, a simple model of the current process estimates that 54% experience a total visit time greater than 30 minutes. Second, this renewal process really does require only address validation and a new photograph. (In reality, there might be other steps like a vision test, but ignore those complications for now.)

We would begin an improvement project on this process by stating the problem we're trying to solve: on days with 50 license renewals, about 54% of customers experience a total visit time greater than 30 minutes. Let's suppose, further, management determines it is acceptable for no more than 10% of customers to experience total visit times more than 30 minutes. This gives us what we need to formulate a simple, measurable goal: On days with up to 50 license renewals, reduce the percentage of visits lasting more than 30 minutes to less than 10 percent.

Having set our goal, next we would observe the process carefully, as it is actually being performed. We would map it out step-by-step, gathering data about how long each step takes, how often a step has to be repeated due to a quality problem, and how many trained staff are available to perform each step in the process. We would also gather data on customer arrivals, the number of license renewals per day, and so on, looking for patterns.

Are there specific hours of the day, days of the week, or months of the year where the problem gets better or worse?

After gathering such data we would analyze them, using a variety of statistical techniques I describe in greater detail later in the book. The goal of such analysis is to provide insight backed by data into the causes of the process problems we are observing.

Suppose that analysis of the data reveals two key sources of our process problem:

- First, we have significant waste in the process.

Obviously the customer is likely to view the long wait as a waste of his or her time. But, the process is also wasting available capacity when the camera sits idle. Employee lunch breaks are entirely predictable, so why does the process break down due to one person's entirely foreseeable absence? This waste is obviously driving part of the excessive wait time.

- Second, customer walk-ins drive most of the variability in the process: the number of license renewals per day varies significantly between 25 and 50, and on high-volume days the wait time explodes.

Because we have gathered the data, we are confident we have identified the most important sources of the problematic process behavior. So now, we brainstorm interventions or improvements we can make to the process to address these two issues. For the sake of concreteness, let's say the project team includes members of the staff who actually perform this work daily. They propose a pair of interventions to address the two issues identified above which they think can be implemented successfully.

- First, the team recommends allowing customers to validate their addresses and make an appointment on the DMV website to renew their license and have the new photo taken.

An appointment system would not eliminate variation in customer arrivals completely, but it would make those arrivals more predictable and allow the office to spread appointments more evenly across the capacity available each day, which directly reduces the average time customers spend waiting for a resource. Further, even if we still allow walk-in customers, the appointment system will tend to limit the total number of customers for this process on a daily basis, which will make the process faster.[1] Further, allowing the customer to populate and submit the required form information online

[1] You're just going to have to take my word for that for now. We'll meet this principle formally in Section 9.10.

removes the need for that step of the process in person, when our timer is ticking.

But our timer measures the visit, not the customer's entire burden. Completing a form at home still takes time, and waiting two weeks for an appointment would not appear in our visit-time measure at all. We should check those burdens too before declaring that we have made the customer's experience better.

- Second, cross-train multiple staff in each office on the use of the camera so that the photography station remains operational during lunch breaks and employee absences.

This intervention helps increase the total capacity of the process because the most constrained resource (the camera) can continue to operate throughout the workday, including during lunch breaks and so on.

Before changing a real office, we can use a discrete-event simulation to estimate what the combined changes might accomplish. Such a model follows individual customers as they arrive, wait for available resources, receive service, and leave, updating the system as each event occurs. The model compares repeated 50-customer days. In the current process, customers arrive randomly, service takes eight minutes on average, and the camera is unavailable for an hour at lunch. In the proposed process, appointments spread the arrivals more evenly, the online form reduces average in-person service time to seven and a half minutes, and cross-trained staff keep the camera operating through lunch.

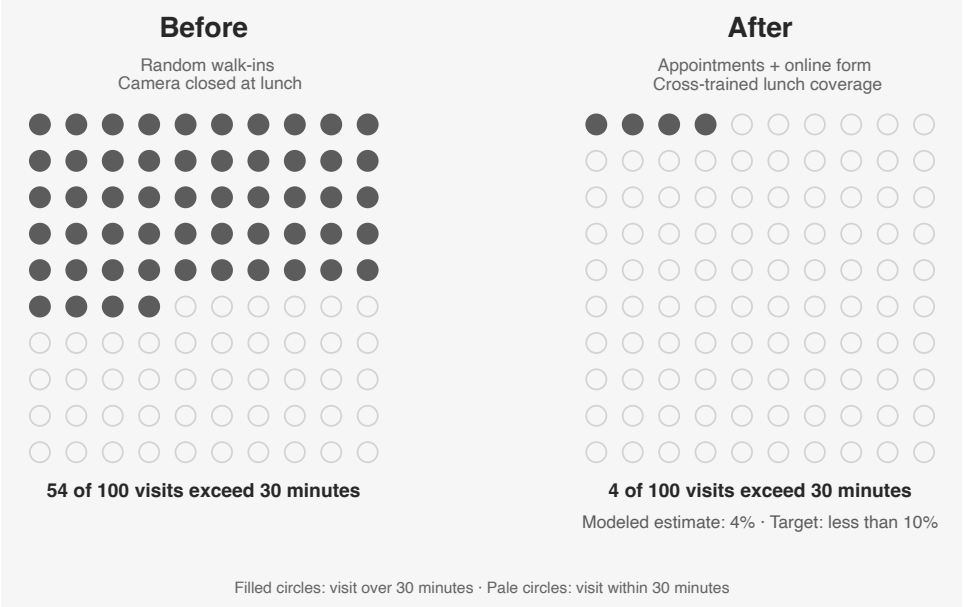



**Figure 1.1.:** Modeled results on repeated 50-customer days before and after the proposed changes.

The model predicts less waiting during the visit and fewer visits exceeding our 30-minute threshold. That prediction is a reason to try the changes, not proof that they will work in the real office. Rather than changing every office at once, the team might pilot these changes at one location, compare the results with the previous process, and revise the design in response to what we learn. Assuming the changes are effective, we could then implement them across the organization. Finally, after implementation, the team would continue to monitor performance to validate that the improved process does in fact reach the goal of <10% of customers experiencing a total visit time more than 30 minutes. If so, then we have achieved our objectives and we can document the improved process and create training materials for staff, action plans for managers, and so on to help sustain the new performance of the process. On the other hand, if we have not hit our goal, we may need to revisit the process and look to identify further improvements. This final monitoring step is crucial to prevent the process from regressing in the future.

In miniature, we have just followed the basic logic of process improvement: we defined the problem, measured the current process, analyzed the evidence, proposed and tested an improvement, and established a plan to control the process afterward. The real world is rarely so tidy, but that basic pattern will guide much of the work ahead.

Let's conclude this illustration by briefly returning to that all-too-human impulse to place blame. I hope this little *hors d'oeuvre* has shown you why

that impulse is counterproductive. I feel frustrated, and I can see these people, so I infer that the people I see must be the *cause* of my frustration. But, from a process improvement point of view, that's very rarely the case. The people weren't the problem; the process was. That's why the first step in process improvement is learning to see the system.

**The Real Secret of Process Improvement.** The people who do and experience the work know things a statistical analysis cannot tell us on its own. That includes both the customers who experience the frustration of the bad process and the staff who have to perform it. In process improvement we value those voices because they are the ones most affected by the process and hence the ones in the best position to tell us how it should work. A premature leap to assigning blame puts people on the defensive and drastically lowers the likelihood of successfully collaborating with those whose insights we must rely upon. Thus, engagement with people is key to successful process improvement.

**The Other Real Secret of Process Improvement.** Listening gives us explanations to investigate; observation and analysis help us decide which ones hold up. The DMV example began with a plausible explanation about lazy staff, but the measured queue and available capacity pointed elsewhere. We need both secrets: access to people's knowledge and a way to check our conclusions.

That, in brief, is what this book is about. *Bon appétit*.

### 1.2. What Is Lean Six Sigma?

People have probably been trying to improve business processes for as long as there have been business processes, but the "scientific" study of management techniques was born in the early decades of the twentieth century. Since then, many authors have proposed, debated, refined, and synthesized different tools and techniques into different process improvement methodologies. All these methodologies seek to carry out the work of process improvement in a structured, systematic, and evidence-based way.

Now is not the time to tell the history of the evolution of those many methods. Instead, I simply note this book uses the *Lean Six Sigma* (LSS) methodology as its point of departure for two practical reasons.

- First, LSS has a deep history, a large practice community, and robust adoption across a variety of domains.
- Second, it gives us a practical way to identify waste and variation, test better ways of working, and make gains stick over time.

In short, this book treats LSS as an established, disciplined approach to process improvement in the real world.

*Lean* and *Six Sigma* were two distinct traditions fused into a single methodology.[2]

- **Lean** traces its roots to the Toyota Production System, developed in Japan in the decades after World War II. At its core, Lean is about identifying and eliminating waste in its various forms: waiting, excess inventory, unnecessary motion, rework, overproduction. Lean offers a set of principles and tools for mapping how work actually flows, spotting where it gets stuck or distorted, and redesigning the process so that value moves cleanly from start to finish.[3]

- **Six Sigma** was developed at Motorola in 1986 in response to quality problems.[4] The methodology then spread elsewhere, notably to General Electric in the 1990s, where CEO Jack Welch touted its positive impacts on the business. Its focus is on reducing variation: the frustrating reality that even well-designed processes produce inconsistent outputs. A process that works correctly nine times out of ten still produces a defect one time in ten, and over thousands or millions of cycles, that adds up fast. Six Sigma provides statistical tools for measuring how much variation exists, diagnosing its root causes, and controlling the process tightly enough that defects become rare. At its most basic, LSS applies the scientific method to improve business processes.

Combined, Lean Six Sigma immediately gives us two useful lenses to inspect a process: look for the waste that slows the work down and the variation that makes its results unpredictable. Although the Lean Six Sigma body of knowledge evolved primarily in the world of heavy manufacturing, today you can find process improvement practitioners applying Lean Six Sigma methods and tools in many service industries, especially healthcare.

For our purposes, the key idea is simple: LSS is a structured, evidence-based way to make work better with broad adoption across a variety of industries. That makes it a natural place to begin teaching process improvement.

But, even though the spine of the book is built around LSS, we will freely borrow tools or techniques from other frameworks whenever we need them. After all, what matters is solving problems, not what we call the tool we use

[2]The integration of the two traditions appears to have begun with George (2002).

[3]The use of the word "Lean" to describe production systems inspired by Toyota originates in Krafcik (1988), which is primarily investigating how a production policy that runs with small inventory buffers can lead to improved throughput while maintaining high quality. Such systems are "leaner" than traditional manufacturing methods which require larger inventory buffers. Womack and Jones (1996) helped popularize the term.

[4]Tennant (2001), p. 6.

to do it or who created it or where it came from. And when the tool we need doesn't exist—whether in Lean Six Sigma or some other framework—we build it.

### 1.3. Who Is This Book For?

This book is for anyone who is willing to commit to the hard work of improving themselves and their business. Process improvement practitioners follow the philosophy of *continuous process improvement*: we strive daily to find better ways to serve our customers and improve our business processes. We want to be responsible stewards of scarce financial resources. We want to be humane managers who value our employees and seek to increase their engagement and satisfaction at work. Learning to do these things isn't easy, exactly. But it's also not some esoteric mystery available only to graduates of a fancy business school. The work begins with careful observation and familiar questions about what happens and why. The methods in this book help us answer those questions reliably. You can learn to do it too if you're willing to put in the time and effort.

Unfortunately, I can't say exactly *how much* time and effort, because I don't know you, your educational background, or your experience with this sort of work. If you have a degree in industrial engineering and 20 years of work experience, I would expect you to breeze through most of the material here fairly easily, at least at the early stages. On the other hand, if you are just starting your career and your education has not been primarily in engineering or a related discipline, you'll need to put more time and effort into acquiring some of the foundational concepts. However, while relevant education and work experience are certainly helpful, they are not necessary. You absolutely can succeed in this field without a college degree in engineering, or anything else for that matter. It is just going to take you a little longer and require a little more effort.

Anything worth learning takes work. That is as true of shaving a few strokes off your golf game or mastering a new piece on the guitar as it is of learning a new set of business tools. Don't worry; it'll be a lot of fun too.[5]

There is another audience I want to identify. This book is also for teachers. I offer these materials to the public under the Creative Commons Attribution 4.0 International License (CC BY 4.0). This means you may freely read, copy, share, change, remix, and otherwise use this content as you see fit, so long as you provide attribution back to the original work here. If you do find the

[5] In fairness, I have been accused of having an idiosyncratic definition of "fun." But just because something's hard doesn't mean it's not fun; quite the opposite. Chess is more fun than tic-tac-toe *because* it's harder.

material useful, or especially if you end up adopting it for a Lean Six Sigma course of your own, please drop me an email at shane@wilkinsadvisors.com. I'd love to hear what you're teaching and which parts of the book help your students. And, of course, typos, bugs, comments, and constructive criticism are always welcome. Please find more details at LICENSE.md. Note that the Creative Commons license only covers content I created—any additional materials that I have cited or included here remain under their original licenses.

### 1.4. Why Another Lean Six Sigma Book?

There are already excellent Lean Six Sigma and process improvement books available. So why another one?

As an instructor, I have struggled to find a single text to give my students that meets three criteria:

- *First*, it should be practical without becoming shallow. Some advanced industrial engineering and operations research texts are mathematically rich but too far removed from day-to-day service work to be accessible to the broad audience interested in process improvement. Others are highly accessible but leave learners unprepared for the jump from introductory material to the more advanced statistical analysis real-world projects frequently require.
- *Second*, the text should function as an actual learning system, not just a reference or certification exam prep manual. My aim is to bridge that gap with a coherent progression of topics, steadily building the mathematical and conceptual foundations but pairing those with practical illustrations and applications. Additionally, such a text requires a sufficient number of exercises, engaging examples, and a structure that works for both self-study and classroom delivery. The goal is understanding and transfer, not rote memorization.
- *Third*, it should be built around open-source tooling and reproducible, scalable workflows. As data and teams grow, our analyses need to remain repeatable and open to inspection. Many common training and industry workflows still rely on proprietary statistical tools, like Minitab or SPSS. Those tools can be useful, but they also create real barriers: high license costs, limited access for students and small organizations, and difficult handoffs when teams use different software.

To reduce these barriers, I have chosen to build the content, examples, and exercises of this book using the Python programming language. Python becomes central at the later belt levels; White Belt deliberately establishes

the method with pencil-and-paper tools and elementary arithmetic. Python combines a relatively gentle learning curve for programming novices with an exceptionally mature ecosystem of free, open-source tools for data cleaning, visualization, statistical analysis, and automation. It is also widely used across universities and industry, and in the data science and machine learning communities, which opens up new possibilities for collaboration and cross-pollination. This means improvement work can be inspected, version-controlled, reviewed in plain text, reproduced by others, and sustained without vendor lock-in. Moreover, the Python skills, scripts, and project artifacts you build during your work in this book are portable. In a classroom, this lowers the cost of learning and makes assignments easier to grade and audit. In organizations, it makes improvement claims easier to validate, maintain, and scale.

So the project is ambitious, but straightforward: provide a practical, pedagogically sound, open-source path for learning and doing process improvement roughly organized around the traditional Lean Six Sigma curriculum. If the book manages to achieve these three goals, I think it will provide a meaningful contribution to the literature. If it helps students and teachers learn to improve real systems with more clarity and less waste, it has done its job.

But the book is also ambitious in a further way.

### 1.5. Beyond Traditional Lean Six Sigma

The familiar Lean Six Sigma toolkit of process maps, control charts, fishbone diagrams, linear regression, and so on, remains essential. But consider some questions our DMV team might still need to answer. How often do customers follow each path through the process? Can another analyst reproduce our calculation of waiting time when next month's records arrive? What happens to the proposed design if demand increases or a second employee is absent?

A process map or a summary statistic can help us ask these questions, but neither can answer all of them alone. We need explicit descriptions of the work, trustworthy records, repeatable calculations, and ways to investigate how changes might interact. This book therefore expands both the tools we use and the methods we teach. Two tools we have created for this work are FLO and `lss4py`. We also give established methods such as process mining and discrete-event simulation a deliberate place in the curriculum.

### 1.5.1. Process modeling with FLO

Imagine that our DMV team has drawn a process map and agreed that it describes the work accurately. We want to preserve that shared understanding, revise it as the process changes, and make its structure available to software as well as people. FLO is a plain-text language for describing a process explicitly: its steps, decisions, roles, resources, and connections. From that description, FLO can check the model's structure and generate diagrams for review.

The advantage is that the description remains available for inspection and reuse. We can record successive versions, compare what changed, and regenerate a diagram from the revised model. If our DMV team changes who performs a task, the model can preserve that change alongside the rest of the process description. Other software can use the structured model for analysis; simulation also requires its own assumptions about arrivals, service times, and resource availability. FLO supplies the process representation, not a simulation engine.

### 1.5.2. Repeatable process calculations with `lss4py`

Suppose the DMV sends us another month's visit records. We should be able to apply the same calculation of elapsed time from arrival to departure, rather than reconstructing it by hand each month. The `lss4py` Python package provides reusable calculations for selected process measures, including elapsed time through a process and the proportion of that time spent on value-adding work. We will meet these measures in more detail as we need them.

Using a shared calculation makes it easier to apply the same definition consistently, rerun an analysis on new records, and let another person inspect the work. Because the package is open-source, its calculation methods can be inspected too. We still have to decide which measure answers our question, check the inputs, and interpret the result. The package's role in this book is to support that work alongside other Python libraries, beginning with these concrete process calculations.

### 1.5.3. Data wrangling and data science skills

Some LSS training curricula simply provide the student with datasets that have already been prepared for analysis. In the real world, the practitioner often has to do this preparatory work. Our DMV records might contain missing departure times, duplicate visits, or different names for the same

activity across offices. Before calculating anything, we need to understand those records and decide how to handle their defects. In practice, improvement work is often blocked not by a lack of analytical tools but by messy, incomplete, or hard-to-access data. This book treats obtaining, cleaning, combining, reshaping, and validating data as part of process improvement itself. Those decisions help determine whether an apparently precise answer deserves our trust.

### 1.5.4. Advanced methods where they are warranted

Some process questions require us to examine how work moves through time, how resources constrain it, and how observed paths differ from the ones we intended. Three methods help us do that:

- **Queueing theory** studies how work waits for service. At the DMV, it helps explain why uneven arrivals can produce long waits even when the office appears to have enough capacity on average. It gives us a way to reason about the interaction of demand, variation, and available capacity.
- **Discrete-event simulation** follows individual cases and resources through a modeled process, as in the DMV illustration above. It lets us explore how appointments and lunch coverage might work together, or how the proposed design behaves under heavier demand, before committing an office to the changes. Its predictions depend on the assumptions we build into the model and must be checked against the real process.
- **Process mining** uses an *event log*: records identifying each case, the activities performed, and when they occurred. If the DMV records these events for each visit, we can investigate whether customers follow the advertised sequence or return to address validation after reaching photography. Mining can reveal recurring paths, repetitions, and differences between recorded work and the intended process. It cannot reveal work the log never captured, so we must check what we find against observation and the knowledge of the people doing the work.

These methods extend the core LSS framework by helping us answer questions that a map or a single average leaves unresolved. Their value lies in the question they help us answer and the evidence they let us examine.

Together, the tools and methods support a connected investigation. FLO makes the process description explicit; `lss4py` supplies selected calculations from observed data; process mining investigates recorded paths; and discrete-event simulation explores proposed changes under stated assumptions. Each contributes a different part of the reasoning, and we must check that the descriptions, records, calculations, and assumptions agree. I think

this is a better way to teach improvement because learners can follow the reasoning from a description of the work to evidence and a proposed change. They can examine the assumptions, repeat the analysis, and revise it when the process contradicts them.

You do not need to learn all of this at once. White Belt begins with observation, pencil-and-paper tools, and elementary arithmetic. Yellow Belt introduces writing FLO models and basic Python analysis; Green Belt deepens independent work with data and reproducible workflows. The planned Black Belt material teaches learners to build simulations and apply process mining, with Master Black Belt emphasizing judgment about their use in wider systems. That higher-belt material is still under development. For now, the point is to see where the foundations lead and why the later tools are worth learning.

## 1.6. Conclusion

The DMV example gave us a first taste of how observation and measurement can challenge an explanation that initially seemed obvious. White Belt asks you to follow that logic through a complete cycle: observe a small process, test a change, and decide what the result warrants. We'll begin with the language needed to describe the work.

Giddy up.

# 2.

# The Method and the Madness

> The medium is the message. — Marshall McLuhan

My main goal in this chapter is to show you the architecture of the book clearly enough that you can navigate it with confidence.

I'll approach the task by describing four things:

- The capabilities the curriculum is intended to develop;
- The three "coordinates" used to organize those capabilities: belt level, knowledge domain, and cognitive depth;
- The learning map created by combining those coordinates; and
- The recurring belt, chapter, and exercise structure used to develop them.

If you are eager to jump right into the content proper, feel free to skip this chapter and return to it later. On the other hand, if you would like to understand the logic before you commit, read on.

## 2.1. A Curriculum Built Around Capability

### 2.1.1. Capability, Not Credential

Ultimately, every curriculum is a kind of hypothesis. If you master *this content*, you should be able to perform *that task* competently. The most important measure for success or failure of an educational program, therefore, is its ability to develop durable professional capabilities.

Of course, it is very difficult to measure or directly evaluate the durability of students' professional capabilities, which puts the credentialing organization in a tough spot. They often approach the task by requiring students to pass a test and then perform a project under the guidance of a mentor. That's

fine as far as it goes; it just doesn't go very far. One successful project does not a subject matter expert make. That's why, if you stick around process improvement circles long enough, you're likely to meet "Black Belts" who couldn't improve a flat tire.

It's a tough nut to crack. But it's not *my* nut. I am not a credentialing body, and I do not propose to certify you as a professional improver of processes or anything else. Not that there's anything wrong with earning a credential; it's just not a game I'm interested in playing.

So reading this book and completing the exercises will not earn you a badge to post to your social media. Instead, what I'm offering you is simply knowledge. In other words, I am trusting you, the reader, to judge your own proficiency with the material and to follow the path of learning wherever your own need and curiosity take you.

But do be aware that the curriculum has a progressive structure: later lessons depend upon the former. So, if you start to get lost, slow down and back up to where you last felt sure on your feet.

There will be ample opportunities for feedback along the way to help you discern whether you're prepared to move to the next topic. Section 2.4 describes those in more detail. However, let us first discuss the specific capabilities the book seeks to develop.

### 2.1.2. Developing Professional Capability

I said every curriculum is a kind of hypothesis about certain professional capabilities. So, let me introduce this curriculum by describing the professional capabilities I aim to develop. Process improvement professionals should be able to:

- **Observe systems honestly.** Improvement begins by looking carefully at how work actually happens, including the workarounds, delays, informal dependencies, and inconvenient facts that an official process description may omit. Honest observation requires curiosity and humility: the process must be allowed to contradict our assumptions about it.
- **Reason operationally.** A practitioner must move beyond slogans and abstractions to ask how a system produces its results through concrete steps, constraints, incentives, resources, and interactions. Operational reasoning connects a proposed change to a plausible mechanism and makes its assumptions visible enough to test.
- **Identify waste and variation.** Waste and variation are not merely vocabulary words; they are recurring ways in which systems consume time, effort, materials, and attention without reliably producing value.

The practitioner learns to recognize them in unfamiliar settings and to distinguish problems created by the design of the system from failures too easily attributed to the people working within it.

- **Collect and interpret meaningful evidence.** Data do not become useful simply because they have been recorded. Practitioners must ask how a measure was produced, what it represents, what it leaves out, how much uncertainty surrounds it, and what conclusions it can reasonably support.
- **Communicate clearly.** Improvement work depends upon explaining problems, evidence, uncertainty, and proposed action to people with different expertise and interests. Clear communication does not conceal complexity, overstate confidence, or bury unwelcome findings; it makes the reasoning available for examination and invites useful disagreement.
- **Lead improvement responsibly.** Process changes affect people, usually in ways we cannot capture in a productivity measure or financial statement. A responsible practitioner asks who will bear the burden of a change, who will receive its benefits, what new risks or unexpected incentives it may create, and whether the people closest to the work have been heard. I do not aim to propose a code of ethics for process improvement professionals.[1] Rather, I aim to cultivate the reflective capacity required for reasonable, prudent, and ethical leadership.
- **Learn continuously from practice and failure.** No method eliminates uncertainty, and no practitioner gets every diagnosis or intervention right. Professional growth requires the discipline to examine failures without defensiveness, revise beliefs when evidence changes, and carry what was learned into the next project. Creating feedback systems that allow us to learn from our mistakes is one of the most important things we can do to cultivate a culture of continuous improvement.

Those capabilities cross disciplinary boundaries. They require process-improvement methods, but also project leadership, data literacy, computing, statistics, and an understanding of the business context in which the work occurs. No single tool, course, or project can establish all of them at once. They have to be developed progressively through explanation, practice, feedback, and increasingly difficult applications.

In defense of the explicit inclusion of the ethical material in our account of these professional capabilities, allow me to observe something. Every mature profession recognizes, in one form or another, that technical capability creates obligations as well as opportunities. This book will not attempt to settle every ethical question a practitioner may encounter, nor reduce responsible conduct to another checklist. Instead, as learners progress through the belt levels, the content will increasingly invite them to notice

[1] The American Society for Quality already publishes a Code of Ethics.

the human consequences of technical decisions, reflect on their own assumptions and incentives, and develop responsible professional judgment proportionate to the stakes. In my view ethics is not something we tack on at the end of the curriculum by promising to abide by a specific professional code. It is rather an intrinsic part of what we are trying to achieve as professionals in improving ourselves and our businesses.

Finally, a clarification about scope. The book will eventually include selected interpersonal and leadership skills material such as facilitation, effective meetings, difficult conversations with stakeholders, presentations, and related subjects. These capabilities matter, and they remain visible in the learning map, but this text is not intended to be an exhaustive soft-skills handbook. Where appropriate, the text will point to additional resources for further depth on these topics.

### 2.1.3. Teaching and Learning This Content

The capabilities required for process improvement mentioned above involve two different kinds of learning. Some of what a practitioner needs can be taught explicitly: formulas, definitions, procedures, principles, and checklists. Other capabilities develop through use. It is one thing to know the formal requirements of a problem statement and another to notice that a technically correct statement is still too broad; one thing to produce a chart and another to recognize that it is answering the wrong question; one thing to collect data and another to judge whether those data are trustworthy.

We can call this second capacity "good professional judgment." It is not an alternative to technical knowledge; it is the ability to use explicit knowledge intelligently in situations that no formula or checklist can specify completely. A capable practitioner must know the tools, but must also learn when to trust them, when to question them, and when to reach beyond them.

This distinction helps explain what is wrong with a naive view that regards Lean Six Sigma as a collection of formulas and templates to be memorized for a certification test. Correctly recalling a definition is not the same as recognizing the relevant problem in an unfamiliar context. Following the steps of an analysis is not the same as determining whether the analysis is appropriate to the subject matter at hand. If we care about capability, instruction must provide opportunities to perform the work, make consequential choices, receive feedback, and try again.

Mentors can be extremely helpful in the learning process because they can observe a learner's performance and provide contextual correction. But mentorship is not the only source of useful feedback, and access to a trustworthy mentor should not be a prerequisite for learning. Worked

examples can expose the decisions behind a successful analysis. Contrasting cases can show why an approach works in one setting and fails in another. Answer discussions can identify what a numerical result does and does not establish. Peers and instructors can challenge assumptions and repeated practice can teach learners to recognize patterns no single explanation could exhaust. This book therefore pairs explicit instruction with repeated observation, realistic cases, guided practice, feedback, and progressively less-structured projects. Early exercises isolate particular concepts and provide substantial direction. Later exercises combine multiple capabilities, introduce ambiguity, and require learners to defend their choices. None of this substitutes completely for experience, but it can prepare learners to benefit from experience and make experience more instructive.

Reflection also has an important place in that process. The book contains journal prompts at the end of every chapter that ask you to pause and record what you noticed, what you assumed, where your reasoning felt uncertain, and how a concept appears in your own work. Reflection is not a substitute for performance or feedback. Instead, its purpose is to make your developing judgment available for examination, so that you can compare earlier reasoning with what later experience teaches you. If you want the material to stick, you have to give the concepts you learn something to grab onto within your own experience. So, please treat the journaling exercises as structural components of the book, not decoration.

## 2.2. The Three Coordinates of the Curriculum

So far we have identified the seven capabilities we're seeking to develop and some pedagogical principles we want to observe in developing our curriculum. Now it's time to actually start identifying what broad kinds of knowledge, skills, and abilities the learner needs to acquire in order to exercise those capabilities. Further, we need to think about what level of mastery is required and in what order to present the topics to provide a logical progression.

The clearest way I can see to do this, which many Lean Six Sigma programs also use, is to build the curriculum around what I will call the three "coordinates" of the curriculum. The *belt level* indicates the practitioner's broad stage of development. The *knowledge domain* indicates the kind of work involved. The *cognitive level* indicates what the learner should be able to do with that knowledge.

### 2.2.1. First Coordinate: The Five Belt Levels

It is customary in Lean Six Sigma to divide the material into various belts, borrowing terminology associated with martial arts in the United States. The belt vocabulary has historical roots and some practical conveniences, even if—to my ear, at least—it sounds kind of hokey. I retain that vocabulary here because it gives learners a familiar way to locate themselves and judge how far they currently want to go.

*Note: The time estimates below refer to studying the instructional material and completing the exercises. They do not include the time required to observe a live process, collect data, implement changes, or complete a real-world project. At Green Belt and above, those activities dominate the schedule, so fixed completion-time estimates become increasingly unreliable.*

- **White Belt** provides orientation, basic awareness, and core terminology. The goal at the White Belt level is to introduce the minimum viable framework that allows a reader with no prior background to complete a small, well-scoped but realistic process-improvement effort. The belt can be completed with pencil, paper, and elementary arithmetic, although calculators and computers are welcome. Reading the instructional prose should take roughly 5–7 hours. Studying the core path and completing a representative selection of exercises should take roughly 12–16 hours; completing the full exercise bank may take 22–30 hours. The small capstone requires additional time spread across repeated daily observations.[2]

- **Yellow Belt** emphasizes practical application of fundamental tools in daily work. It is a useful target for a small-business owner, an early-career employee, or anyone seeking near-term improvements without yet committing to leading a dedicated project team. Completing the material and exercises should take roughly 25–35 hours.
- **Green Belt** marks the transition into independent project execution and intermediate analytical capability. The learner begins to lead structured improvement projects, use a more substantial statistical toolkit, and develop the facilitation and project-management skills needed to move work through an organization. This is a serious step up from Yellow Belt and will usually take several weeks of study.
- **Black Belt** is intended for advanced practitioners, many of whom perform process improvement as a major part of their professional work. The material approaches upper-division undergraduate or early-stage

[2] Note that our White Belt requires *considerably* more time and effort than other curricula, which treat it as essentially a one-hour overview of the content to be learned in Yellow Belt.

graduate rigor. Black Belts tackle more complex analytical problems, adapt or construct tools when necessary, and begin coaching Yellow and Green Belts. This material will usually require months rather than weeks to master.

- **Master Black Belt** shifts the center of gravity from analytical technique toward enterprise program design and strategic governance. The central questions concern how to build, coordinate, evaluate, and sustain organizational capacity for learning and improvement. This level is aimed primarily at process improvement leaders who work across business functions and interact regularly with senior executives. The time commitment to complete the material at this level is highly variable and dependent on prior experience.

These levels are cumulative but not perfectly linear. An experienced analyst may be advanced in statistics and still need to revisit foundational process-mapping skills; an experienced manager may understand organizational governance but need substantial work in data analysis. Use the belts to locate the next useful stretch of the journey, not to make a global judgment about your worth or experience.

Again, this book uses the familiar belt vocabulary because the levels help learners orient themselves, return to foundations when needed, and keep moving.

Finally, we should mention that although these five belt labels are fairly common across different Lean Six Sigma curricula, their contents are not uniform. Some authors put a certain topic in Yellow Belt, others defer the same topic for Green Belt. This book, for instance, makes White Belt a much more substantial piece of the curriculum than others in order to make room in Yellow Belt for more instruction in Python.

### 2.2.2. The Second Coordinate: The Six Knowledge Domains

Belt level tells us how far a practitioner has progressed, but it does not tell us what kind of capability is under discussion. Our second "coordinate" helps structure that dimension of the curriculum by dividing the material into six broad knowledge domains:

Lean Six Sigma Principles and Tools contains the process-improvement toolkit proper: Lean principles, PDCA, DMAIC and A3 structure, process mapping, waste identification, and the core quality tools.

Leading and Managing Projects contains the capabilities needed to move from having a good idea to running an improvement effort well: problem

framing, goal setting, project scoping, facilitation, stakeholder communication, and project cadence.

Data Understanding contains the habits of mind needed to make sense of process data in the first place: measurement concepts, data types, cleaning and validation, summarization, visualization, and exploratory interpretation.

Computational and Data Systems contains the software and systems capabilities that make modern analysis reproducible and operationally useful: Python workflows, notebooks and scripts, environments and dependencies, SQL and databases, version control, testing, automation, and reusable tooling.

Statistical Analysis contains the quantitative methods used to move from description to evidence-based judgment: descriptive statistics, probability and distributions, hypothesis testing, modeling, and experimental methods.

Business Understanding contains the organizational and economic context that keeps improvement work tied to reality: customer value, financial framing, strategic alignment, and enterprise-level governance.

These domains are practical buckets for curriculum design, not rigid silos. They overlap constantly in real projects, and the same tool may plausibly fit in more than one. It doesn't really matter at the end of the day, as long as we know what the tool is for and can recognize when to use it.

Further, each domain appears across multiple belt levels, but its emphasis and depth change. Work in the Computational and Data Systems domain, for example, begins with simple exercises teaching programming basics but grows toward automation, testing, and reusable systems; business understanding develops from basic financial framing toward strategic alignment and enterprise governance. But we're not trying to turn you into an accountant or a software engineer. In each case, we keep the scope manageable by tying the content we teach to the requirements of a successful process improvement professional. The domains let us see those strands of development without pretending they occur independently.

Attentive readers may have already observed that the domains don't align neatly with the seven capabilities. Isn't this all getting a little baroque? Well, yes and no. Yes, there is some complexity to it. But there's a point to that complexity. Capabilities like clear communication or responsible leadership are by their nature broad as well as highly contextual, which is helpful to know but nearly impossible to operationalize in a curriculum. The knowledge domains help us bucket the kinds of knowledge those capabilities require. So no, it isn't more complex than it needs to be, at least as far as I can tell.

### 2.2.3. The Third Coordinate: Describing Cognitive Performance

The final coordinate describes not just *what* is taught at each belt level, but *how deeply*. For that, we need a shared vocabulary for cognitive performance, and this book borrows one from educational psychologist Benjamin Bloom's influential taxonomy of educational objectives.

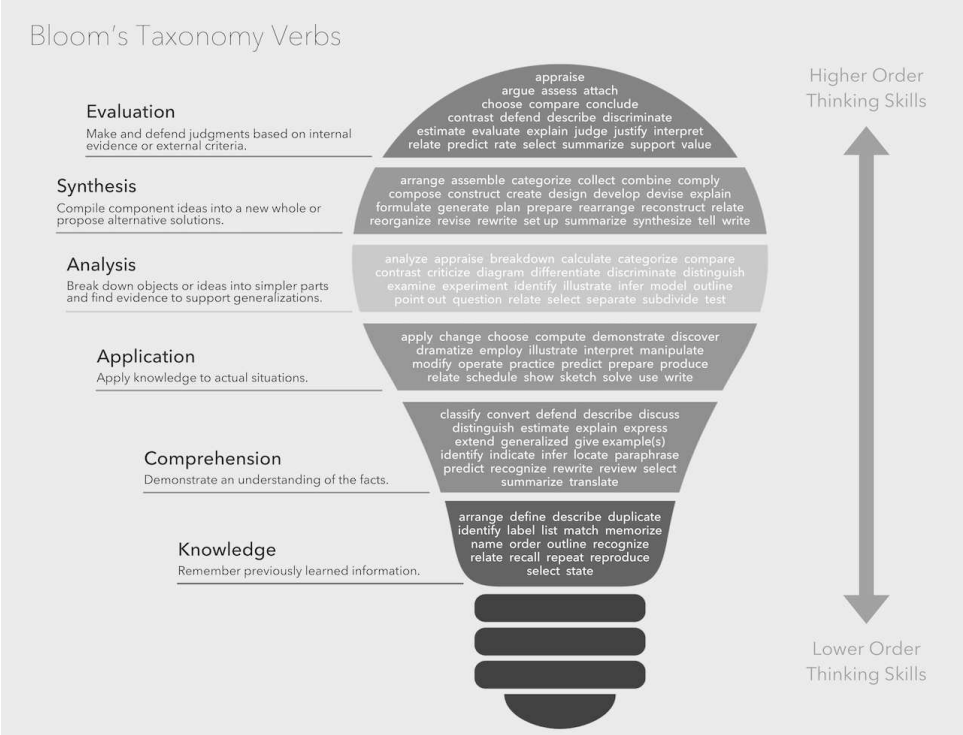



**Figure 2.1.:** Revised Bloom's Taxonomy of Educational Objectives

> **Media Credit**
>
> Source: File: Blooms Taxonomy Verbs.png, Wikimedia Commons.
> Creator: Fractus Learning.
> Rights: CC BY-SA 4.0. Used under license (not fair use).
> Modification: No substantive modification; resized by the publication renderer.

- **Remember** Recall basic information such as definitions, formulas, and core concepts.
- **Understand** Explain ideas, recognize examples, and restate a concept in one's own words.
- **Apply** Use a concept or procedure in a concrete situation.
- **Analyze** Distinguish relevant features, compare alternatives, identify patterns, and investigate causes.

- (Evaluate) Judge the adequacy or usefulness of an approach and defend a conclusion with evidence.
- (Create) Integrate ideas to design a new analysis, intervention, model, or explanatory framework.

I use these levels pragmatically, not as an ironclad account of how people actually think and learn. They are shorthand for the kind of performance I expect learners to demonstrate at a particular point. Their value is that they allow me to make these goals and objectives specific, chapter by chapter.

Further, allow me to caveat that Bloom's Taxonomy is often taken to imply a kind of hierarchy or progression that makes "observation" somehow a less demanding or less impressive cognitive achievement than synthesis or evaluation. I agree with biologist Maura Flannery that observation "is often difficult and complex, a skill that needs to be learned and a talent that is much more developed in some" than others (Flannery 2007, 561). Process improvement relies heavily upon observation, and learning to see is one of its primary challenges.

### 2.3. Putting It Together: The Learning Map

So, now we have five belt levels, six knowledge domains, and six broad levels of cognitive performance. All that remains is to make a matrix and begin selecting topics and bloom levels by belt. The Learning Map puts those coordinates together in a table showing what each belt teaches in each domain and what learners are expected to do with it. The detailed accounting lives in an appendix:

- Lean Six Sigma Learning Map

The map is intentionally coarse-grained. Its purpose is to show the main capability bands, not to reproduce every chapter-level Learning Objective one-for-one.

Think of the appendix document as a navigation chart rather than a checklist. If you are self-studying, use it to identify your likely next bottleneck.

Real projects are messy, and development is not always linear. You may need to pull a Black Belt-level statistical tool into a Green Belt project or revisit White Belt fundamentals halfway through a difficult initiative. That's not failure; that's just good applied practice.

### 2.4. From Learning Objectives to Practice

Understanding Bloom's taxonomy makes it much easier to design opportunities for practice and self-assessment. Each chapter begins by identifying one or more Learning Objectives that state what the learner should be able to do with the content. Each learning objective is aligned to a specific knowledge domain and Bloom level. The exercises at the end of the chapter are then deliberately designed to test that content at the appropriate level.

At the White and Yellow Belt levels, four recurring exercise types provide much of the structure:

- **Recall and Explain** targets *Remember* and *Understand*. The learner supplies a definition or explains a concept in their own words. These are low-stakes exercises, but they build the vocabulary needed for more complex work.
- **Apply to a Scenario** targets *Apply*. The learner receives a miniature case or worked problem and uses a concept or tool. At White Belt this may be pencil-and-paper; at Yellow Belt and above it often involves Python.
- **Analyze and Identify** targets *Analyze*. The learner examines a short description of a process or situation and diagnoses what is happening—classifying wastes, identifying plausible causes, spotting gaps in a process model, and so on. It is easy to define a form of waste; it is harder to recognize one in an unfamiliar context.
- **Reflect and Transfer** asks the learner to connect the chapter's ideas to their own experience or workplace. Depending on the prompt, this may involve application, analysis, or evaluation. Because these questions rarely have a single correct answer, answer guidance describes the features of a strong response rather than prescribing a model answer.

Short **Journal and Reflect** prompts appear alongside these exercises throughout the book. They capture what the learner noticed, assumed, or still finds uncertain. They are a metacognitive practice rather than a separate rung of Bloom's taxonomy. That just means their purpose is to make us think about our own thinking. The journal preserves earlier reasoning so we can revisit it after more experience.

This pattern is consistent enough that engaged readers should recognize it quickly, but flexible enough to follow the content rather than forcing every chapter into the same mold. Not every chapter will contain every exercise type, and some will emphasize one more heavily than another. The exercises form a practice bank rather than a demand every learner complete every item. Independent learners and instructors may choose to complete a representative subset when time is limited, provided the selection still

covers the chapter objectives and includes the integrative exercises that feed the belt project. For a reduced White Belt path, preserve the Wash n' Fold case work and at least one transfer exercise in each chapter.

At the **Green Belt and Black Belt** levels, the cognitive demand shifts upward and the exercises increasingly require interpretation, evaluation, and design. Pencil-and-paper work gives way to substantial data work in Python, and a correct numerical answer is no longer sufficient. Three additional exercise types become prominent:

- **Compute and Interpret** gives the learner a dataset and asks them to produce an output using `lss4py` or standard Python libraries, then explain what it means for the process under study. Getting the number is table stakes; the real work is the interpretation.
- **Critique and Evaluate** presents a completed analysis—a control chart, regression output, FMEA, process model, or similar artifact—and asks the learner to assess its quality, identify what it does and does not establish, or recommend a next step.
- **Design** asks the learner to plan something from scratch: a sampling strategy, a measurement system, an experiment, or a DMAIC project outline. These are among the most demanding exercises in the book and appear most frequently at the Black Belt level.

At the **Master Black Belt** level, the center of gravity shifts toward open-ended reflection and program-level design. Exercises ask for deployment roadmaps, coaching responses to realistic project scenarios, and strategic evaluations of program-design choices. At this level there are rarely unique correct answers; there are more and less defensible ones.

If you are reading this book at modernprocessimprovement.com, then you are reading the HTML version of the book, which means exercises with a clear correct answer include a collapsed answer callout. On the other hand, if you are reading the printable PDF, those authored answers are collected in the solutions appendix instead of being repeated after each exercise. Open-ended White Belt chapter and capstone exercises include acceptance guidance about what to look for rather than one prescribed answer. These are public practice exercises, separate from the operational chapter reviews and cumulative White Belt assessment delivered through the LMS. The operational assessment questions, scoring keys, and feedback are not included in the PDF or offline package.

Some chapters also include one or two optional stretch prompts, marked with a visible **[Stretch]** label. These extend a concept, foreshadow a later capability, or give ambitious learners a glimpse of what becomes possible at

a higher belt. They are motivational prompts, not hidden requirements for demonstrating the chapter's expected competence.

## 2.5. Anatomy of a Belt and Its Chapters

The exercise patterns just described sit inside a larger instructional structure. The individual chapter is important, but the belt level is the primary unit of development in this book. Each belt forms an arc: it opens with a problem that creates a need for the new capabilities, develops those capabilities across a sequence of chapters, and closes by asking the learner to integrate them in a capstone project.

### 2.5.1. The Threshold Story and Belt Promise

Each belt begins with a *Threshold Story*. The story places the learner at the threshold of a new level of practice by presenting a grounded business problem that cannot yet be solved responsibly with the capabilities developed so far. It includes incomplete evidence, competing explanations, failed or fragile improvement attempts, and recognizable human stakes. There is no heroic leader with a magical solution. The point is to create an honest reason to learn the methods that follow.

For example, White Belt opens in Chapter 3 with a Threshold Story in which a laundromat struggles with recurring late Friday pickups. The staff are busy, different people have plausible but conflicting explanations, and a small improvement appears to work before the process drifts backward. The story does not solve the problem for the learner. It establishes the situation that the learner will gradually become capable of investigating.

The Threshold Story also serves as the belt's instructional on-ramp. It identifies:

- The learner's assumed starting point;
- The *Belt Promise*;
- The anchor case that will recur throughout the belt;
- The capabilities the belt will develop;
- Any required equipment or technical setup;
- The artifacts the learner will build; and
- The capstone in which those artifacts will be integrated.

The Belt Promise states what the instructional system should enable the learner to do in practical terms. White Belt, for example, promises that by

the end of the belt the learner will be able to run one complete, evidence-based improvement cycle on a small real process and communicate the result clearly, using only pencil-and-paper tools if desired.

Learners completing a Belt level will be asked to perform an honest self-assessment about whether they feel that promise has been met. Ultimately our goal is to collect this feedback from our learners via the modernprocessimprovement.com website to help improve the book in the future. In essence, what we are trying to do with the website is to apply process improvement thinking to the process improvement curriculum itself.[3]

### 2.5.2. The Instructional Chapters

After the Threshold Story, the belt proceeds through a sequence of instructional chapters. The anchor case introduced in the story runs through those chapters, allowing the learner to return to the same process with progressively more capable eyes. Each chapter contributes concepts, methods, and usually an artifact that will be needed later in the capstone. Non-anchor examples and transfer exercises keep the learner from merely memorizing the details of one case.

Although the subject matter varies, instructional chapters have a recognizable shape:

1. **Why This Chapter Matters** establishes the practical problem and explains why the chapter belongs at this point in the belt.
2. **Learning Objectives** state what the learner should be able to do, identify the relevant knowledge domain, and specify the expected level of cognitive performance.
3. **Concepts, examples, and worked applications** introduce the material in context and expose how it is used.
4. **The anchor-case thread** connects the chapter to the belt's continuing problem and to an artifact needed for the capstone.
5. **Exercises** provide objective-aligned practice using the anchor case, additional examples, and unfamiliar situations requiring transfer.
6. **Answer guidance** provides feedback appropriate to the exercise. HTML places authored answers in collapsed callouts; the printable PDF collects those same answers in a solutions appendix.

[3] Once I hoped this idea of applying process improvement tools and techniques to improve the process improvement curriculum might be a genuinely original idea. Fortunately, it is not so. (Bryk et al. 2015) presents recognizable process improvement concepts as part of an extended argument about how to improve the quality of the U.S. educational system generally, and explicitly credits the idea to process improvement thinking encountered in a healthcare context. May their tribe increase!

7. **Reflection prompts** ask the learner to examine assumptions, uncertainty, and connections to their own experience.
8. **The Chapter Summary** restates the durable ideas and makes the transition to the next part of the belt visible.

Not every chapter gives these elements equal weight. A conceptual chapter may emphasize definitions, distinctions, and reflection; a computational chapter may emphasize worked code examples and interpretation; a project chapter may require the learner to produce and defend an artifact. But the shared structure provides orientation and accountability without forcing unlike subjects into identical containers.

### 2.5.3. Artifacts and the Capstone

The chapters are designed to accumulate work rather than produce a series of disconnected quiz scores. In White Belt, for example, the learner works through the Wash n' Fold problem definition, observation record, Mapping Worksheet Set, current-state SPPM, Baseline Performance Readout, waste diagnosis, countermeasure design, before-and-after comparison, and Act recommendation as the case unfolds. Those completed artifacts provide a worked model for the capstone. The learner then constructs the same chain of evidence for a small daily process at home: what was happening, what evidence was collected, what changed, what did not, and what should happen next.

The capstone is therefore the principal integration point for the Belt Promise. It does not prove a learner will succeed in every future project, but it requires more than recalling terminology or applying one isolated tool. It asks whether the learner can coordinate the capabilities developed across the belt and use them to produce a defensible piece of improvement work.

## 2.6. How to Use This Book

If you are studying independently and have no prior Lean Six Sigma background, begin with White Belt and proceed in order. If you have previous training or experience, do not assume the color printed on a certificate identifies the best entry point. Use the learning map and chapter objectives to locate the earliest capability you cannot yet demonstrate confidently, then begin there. Moving backward to rebuild a missing foundation is not regression; it is efficient learning.

Once you have chosen a belt, begin with its Threshold Story. Read the Belt Promise carefully, note the artifacts you will be expected to produce, and

keep the opening problem in mind as you work through the instructional chapters. The chapters are sequenced deliberately, and later chapters may assume you have completed the exercises and artifacts introduced earlier.

Do the exercises before consulting the answer guidance. In the online version, open a collapsed answer callout after making your attempt. In the printable PDF, consult the solutions appendix. Use those materials to examine your reasoning, not merely to determine whether your final answer matches. For open-ended questions, compare your response with the stated criteria for a strong answer and note what you overlooked or would change.

Keep your journal entries and chapter artifacts together as you progress. The journal gives you a record of how your judgment develops; the artifacts become inputs to the belt capstone. When you reach the capstone, complete it before deciding whether you have fulfilled the Belt Promise. Use any gaps you discover to identify what to revisit and what your next useful stretch should be.

## 2.7. Conclusion

You now have the full floor plan. The belt levels tell you how far to go; the knowledge domains tell you what the work covers; cognitive levels make the expected performance explicit; the learning map shows how those dimensions intersect; and the exercise structure turns the architecture into practice.

None of this needs to be memorized. Return to this chapter when you need a compass bearing, and treat the learning map as a working reference you can consult as you progress.

Enough pedagogy; let's build something.

## Part II.

# White Belt

# 3.

# Threshold Story: The Friday Pickup Problem

A laundromat owner in a college town hears one recurring complaint: “You said after 5, but my order is not ready.” No dramatic failure is visible. The washers wash. The dryers dry. And the staff are consistently busy. But Friday pickup delays keep happening and online ratings are drifting downward. The owner feels pressure to offer discounts just to calm people down, cutting profit margins uncomfortably thin.

The owner has invited our team to come help figure out what the problem is. Different team members have different theories. One says folding is too slow. Another says intake is the real problem. A third says the issue is not speed at all, but release timing and queue buildup. They cannot agree, and each person has an anecdote or two that sounds convincing. Each feels strongly about their pet theory, but they cannot agree on what evidence would allow them to choose among the explanations.

At the staging rack, a worker lifts a bag, reads its tag, and puts it down again. The last digit could be a three or an eight. The worker returns to the ticket before writing a replacement tag; meanwhile, a customer at the counter is waiting for an answer. Other bags are ready, but their pickup notifications will not go out until the next notification batch. From the counter, the customer can see busy staff and racks of laundry, but cannot tell whether their own order is among them. The worker has already apologized several times this afternoon.

When the team experiments with a simple visual queue at folding, late pickups drop for two days. Then performance drifts back. The change seemed to help, but it did not stick, and it is not clear that it was a sufficient solution in the first place.

Our White Belt content exists for exactly this kind of situation. We do not need a heroic turnaround leader, and we almost certainly do not need an expensive automation system. What we need is a disciplined methodology: define scope, observe directly, map what actually happens, measure baseline flow, identify waste, test a practical countermeasure, and evaluate results. That's what we're going to learn.

### 3.1. Learner Starting Point

There are no prerequisites for White Belt. The only mathematics you need to know is elementary arithmetic: addition, subtraction, multiplication, and division. Everything else we'll explain as we go. If you'd like to rebuild or refresh that foundation, see *The First Reckoning: What Did You Make One?*, the optional arithmetic appendix.

### 3.2. Belt Promise

By the end of White Belt, you will be able to run one complete, evidence-based improvement cycle on a small real process and communicate the result clearly. You can complete the belt with a notebook, printed worksheets, and elementary arithmetic. Feel free to use a calculator, spreadsheet, or other computer tool if you wish, but none is required.

### 3.3. Why This Case? Why Now?

Wash n' Fold is a fictional case constructed to resemble an ordinary small service business. It has queues, handoffs, rework risk, and competing priorities, just like many office, service, and operations workflows. That makes it a good proving ground for techniques you can transfer later to your own environment.

### 3.4. What You Will Learn in This Belt

In the upcoming White Belt chapters, you will learn to:

- Ground improvement work in Lean foundations and customer value.
- Use PDCA to structure a first improvement cycle.
- Build a current-state process map from direct observation.
- Calculate baseline flow metrics and interpret bottlenecks.

- Diagnose waste and apply practical 5S countermeasures.
- Integrate results into a capstone-quality improvement narrative.

### 3.5. Required Equipment

Assemble your materials before beginning so you don't have to interrupt an observation to find them. As you work through White Belt, you'll need:

- A notebook, loose paper, or digital notes space for observations and reflections.
- A clock, watch, or phone timer for quick run-time sampling.
- A basic calculator if you want.
- Printouts of some templates we will provide later; a spreadsheet capturing the same data is an optional alternative.
- A process you can observe with clear start and stop events.
- About 60 to 90 minutes of uninterrupted working time per study session.

### 3.6. White Belt Artifact Roadmap

These are the core artifacts you will encounter in the guided Wash n' Fold case and then build from your own capstone evidence:

1. **Problem definition:** a problem statement and SMART goal.
2. **Observation record:** an event-log excerpt containing the timestamps and context behind the analysis.
3. **Mapping package:** a Mapping Worksheet Set and a verified current-state SPPM.
4. **Baseline Performance Readout:** clearly bounded measures of customer and process performance.
5. **Diagnosis and trial plan:** an evidence-linked waste diagnosis and a bounded countermeasure with measures and ownership.
6. **Check and Act record:** a before-and-after comparison and a recommendation to standardize, revise, abandon, or retest the change.

The Wash n' Fold case sometimes supplies a completed artifact when reproducing it in full would add clerical work rather than useful practice. In those cases, you will reconstruct a bounded portion, inspect the complete reference, and use it in the next decision. The capstone asks you to create the corresponding artifacts from your own smaller process.

### 3.7. Capstone Preview

In the White Belt capstone, you will use the completed Wash n' Fold artifacts as models while you run the same cycle on a small process at home that you perform every day. The capstone asks whether you can transfer the method to real work whose answer the book has not supplied. What matters is completing the cycle honestly and learning from the result, whether the intervention succeeds or not.

The next chapter starts the method directly: it introduces the first principles of Lean, customer value, and process thinking that make the rest of the belt actionable.

# 4.

# Lean Foundations

> "A dissatisfied customer does not complain; he just switches." — Oliver Beckwith, quoted by W. Edwards Deming

## 4.1. Why This Chapter Matters

Success in business depends on many factors, including some nobody can control. Effort, talent, and a sound business plan are all vital components. But organizations also need processes capable of producing the quality their customers expect. An organization can fail despite serious effort when it misunderstands what its customers value or how their processes create that value.

This chapter introduces four key concepts—firm, customer, value, and process—and two guiding principles that ground improvement work: continuous improvement and respect for people. Together these six ideas provide a foundation for the improvement work that follows. Without that foundation, later tools can become rituals rather than the means of improving performance.

The next chapter turns this vocabulary and philosophy into actionable guidance.

## 4.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Business Understanding Define "firm," "customer," "value," and "process," and explain how they are related **Understand**

- Data Understanding Explain what a process run is and why multiple runs matter for observing and improving a process **Understand**
- Data Understanding Define a process variant and explain why documenting variants matters during process observation **Understand**
- Lean Six Sigma Principles and Tools Describe continuous improvement and respect for people as mutually reinforcing foundations of Lean practice, and explain why neither is sufficient without the other **Understand**
- Lean Six Sigma Principles and Tools Recognize practices that support—or undermine—continuous improvement and respect for people in an organization **Understand**

## 4.3. Foundational Concepts

Let's begin with four interrelated concepts: the firm, the customer, value, and process. These concepts are absolutely crucial to everything we will do later. Think of this section as a primer on the language of process improvement.

### 4.3.1. The Firm

**Definition 4.1.**

> **Definition: Firm**
>
> A firm is an organized entity that coordinates people and resources to provide goods or services of value to customers.[1]

Note how broad this definition is. A solo lawyer, a local coffee shop, and a global auto manufacturer are all firms in this sense: each exists to deliver value to someone. Firms differ widely in scale and industry. One thing they all have in common, however, is that they all have customers. Depending on the industry, customers may also be called clients, patients, students, patrons, members, or users.[2]

So let's get clearer on what a customer is.

[1] This book uses *firm* as an analytical term drawn from economics and organization theory for an organized unit that coordinates resources to produce goods or services. The usage is broader than ordinary commercial speech: when the same organizational questions arise in a public or nonprofit body, the book treats that body as a firm for purposes of analysis.

[2] These labels identify the recipient of the benefit, not necessarily the funding party. A student, patient, or user may receive a service that a family, employer, insurer, or government pays for. When the recipient and funding party differ, this book reserves *customer* for the party receiving the benefit or use of the firm's goods or services.

### 4.3.2. The Customer

**Definition 4.2.**

> **Definition: Customer**
>
> The customer of a firm is the person or organization that receives the benefit of its services or the use of its goods, regardless of who pays. The customer, payer, purchaser, and funder may be the same party, but they need not be.

The customer should be distinguished from other kinds of stakeholders, such as employees, shareholders, suppliers, regulators, and community members. A *stakeholder* is someone affected by the firm and its operations. All customers are stakeholders, but not all stakeholders are customers. The customer is a special stakeholder because the customer receives the benefit of the goods or services the firm provides. In many commercial settings, the customer also pays directly. In healthcare, education, insurance, government, and family purchases, another party may pay or authorize funding. Keeping the recipient and funder distinct prevents the source of payment from silently replacing the needs of the person or organization the process exists to serve.

You will also sometimes see people distinguish *internal* and *external* customers. Large firms are often divided into functions—finance, HR, operations, sales, and engineering—that serve one another. In this way of speaking, the downstream business function consuming the upstream business function's output is an *internal customer* of the upstream unit. I find this terminology somewhat misleading. Fulfilling a request from an internal customer does not directly create value for the people the organization exists to serve and no external party is willing to pay the firm for fulfilling that internal request itself. To put the point bluntly: if nobody is buying what we're making, we don't need to hire employees, file taxes, or organize a training retreat, as important as those activities may be.

Of course, firms still need their internal functions to work together efficiently and effectively, and process improvement projects often do focus their efforts on those internal functions. In that setting, it can be useful to treat the downstream recipient of an internal service *as if* it were a customer. To keep that "as if" qualification visible, this book refers to such recipients as *internal customers*. Unless otherwise qualified, *customer* means the external recipient of the benefit or use of the firm's goods or services.

> **!** Question
>
> Who are your firm's customers? Who is NASA's customer, and how does that customer fund NASA?

But what is this "value" our customers want?

### 4.3.3. Value

**Definition 4.3.**

> **i** **Definition: Value**
>
> The value of a product or service is its worth to the customer, judged from the customer's point of view.

When I order a pizza, I value an enjoyable meal. When I hire a financial advisor, I value credible guidance toward my financial goals. When I buy a car, I value safe, reliable transportation. When I join a gym, I value convenient access to a suitable place to exercise.

One of the central challenges in process improvement is maintaining a clear view of what customers value. Changes made without that understanding can reduce the benefit customers seek. If that happens, customers may leave and the business may fail.

Let me close the discussion of value with an important clarification: value is not the same as price. Price is the amount the seller charges; value is the customer's judgment of the benefits received. In a commercial setting, a purchase is attractive when the customer expects those benefits to justify the price and other costs, such as time, effort, and risk. Because those judgments differ among customers and change over time, price is just one piece of evidence about value, not an objective measure of it.

Customers who consistently receive good value are more likely to continue funding the firm and recommend it to others. Customers who do not may withdraw their support or look elsewhere when alternatives exist.

Now let's tease out a corollary from that definition of value.

> **i** **Corollary: Customers Define Value**
>
> The customer is the one who defines value, not the firm.

Firms cannot declare their output valuable by fiat. Customers judge value through their needs, choices, and experience, whether their payment is direct, indirect, or collective.

> **!** Question
>
> Why is the business portrayed in this comic struggling?

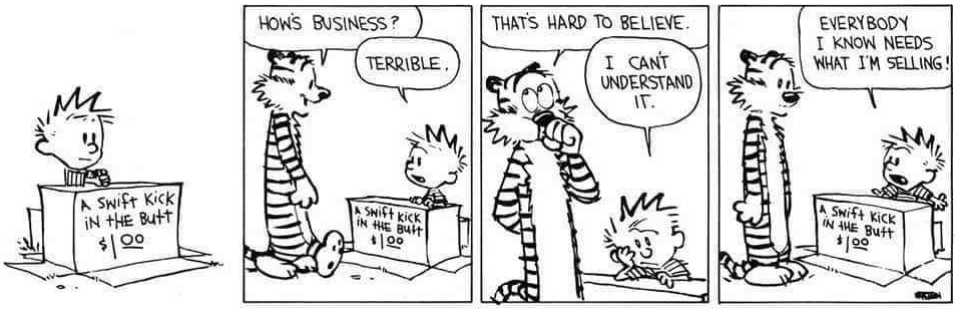



**Figure 4.1.:** *Calvin and Hobbes* by Bill Watterson.

> **i** **Media Credit**
>
> Source: *Calvin and Hobbes*, July 31, 1992, originally published by Universal Press Syndicate. Creator: Bill Watterson.
> Rights: Used under fair use for educational commentary and analysis.
> Modification: No substantive modification; resized by the publication renderer.

> **Answer**
>
> The business is trying to specify value from the point of view of the owner, not the customer.

The example in the comic is exaggerated, but losing sight of how the customer defines value is a common problem.

We need one more foundational concept to link the other three concepts together. Consider how an ordinary firm delivers what its customer wants.

> **!** Question
>
> Choose a firm you know—a café, repair shop, or delivery service, for example. List the steps by which it delivers one product or service to a customer.

Answer

For a café, the steps might include taking an order, preparing the drink, and handing it to the customer. Your sequence will depend on the firm and the product or service you chose. We call such a sequence a **process**.

#### 4.3.4. Process

Let's now give a definition of a business process.

**Definition 4.4.**

**Definition: Business Process**

A business process is a set of steps designed to produce a specific product or provide a specific service to a customer.

Note

Our definition above is specifically about business processes, as opposed to chemical, geological, biological, or some other kind of process. However, for brevity, we will usually just say "process" unless the broader context makes it necessary to clarify.

For example, if the firm manufactures tires, it provides value to its customers by performing a process whose output is a tire that meets the customer's specifications.

While we're at it, let's also define a process step.

**Definition 4.5.**

**Definition: Process Step**

A process step is a discrete unit of work.

Process steps may involve different people, tools, and information. A process can cross organizational boundaries or return to the customer for action. For example, hiring may begin with a manager's request, move through HR, and return to the hiring manager for interviews and selection. One process can also feed another; hiring feeds employee onboarding, for example.

The point is that a process is built from discrete steps that can be observed and improved.

### 4.4. Process Runs and Process Variants

While we're talking about processes, let's introduce two more concepts: a process run and a process variant.

**Definition 4.6.**

> **Definition: Process Run**
>
> A process run is one complete instance of a process, traced for one thing from the process's defined start event to its defined stop event. That thing might be one customer order, one patient visit, one service request, one batch of work, or one invoice.

For example, if you're analyzing a customer service process, one process run would be:

- customer calls (start)
- agent answers
- agent resolves the issue
- call ends (stop)

That single journey is one run. If you watch five different customer calls on a Monday morning, you have seen five runs. Process runs become important when we begin observing a process. A single run cannot show whether what we saw was typical. Because any one run may be unusual, we usually observe multiple runs and compare their paths, timing, and outcomes.

The second concept we need here is the process variant.

**Definition 4.7.**

> **Definition: Process Variant**
>
> A process variant is a recurring alternative path within the same process boundary. Variants share the same start event, stop event, and primary unit of analysis but differ in one or more steps, roles, timings, or decision paths.

Some variants are intentional. Different products, customer needs, or service levels may require different standardized paths. Think of a car going down the assembly line: some times we want to produce the base model, sometimes we want to produce a higher trim level that has fancier seats, or a better radio, or some pricey doodad. Producing a higher trim level vehicle might be a legitimate process variant that requires slightly different work than the base model. More commonly, however, process variants can often reveal inconsistent work, unnecessary exceptions, rework, or departures from the intended process.

The goal is not to eliminate every variant merely because it differs from the primary path. The goal is to distinguish necessary, standardized variants from unintended or wasteful ones, improve each necessary path, and remove avoidable variation.

Now that we've got a clear understanding of the definitions of the key concepts of process improvement, let's turn now to the philosophical foundations to complete this stage-setting chapter.

## 4.5. Two Pillars

The Toyota Way is a distillation of the organizational principles that have led the Toyota Motor Corporation to become the largest automobile manufacturer in the world.[3] The Toyota Way explicitly names two principles "pillars" of its approach to business: continuous improvement and respect for people. In this book, we use them as philosophical and ethical foundations for Lean practice. Neither works well alone. Improvement without respect can become exploitation, while respect without disciplined improvement leaves preventable problems in place, frustrating customers, staff, and management alike.

### 4.5.1. Respect for People

Respect for people isn't just about being polite. Respect for people means recognizing them as agents with knowledge, dignity and the same rights we want for ourselves. Of course, we cannot have respect for people in a work environment characterized by prejudice or discrimination. But meeting the bare requirements of labor law is not yet to begin the deeper work of building a culture of respect.

[3]The Toyota Way is also the subject of an excellent book of the same name by Jeffrey Liker.

- We respect customers when we view them as people who have the right to make their own, informed decisions and provide them with honest information; we don't try to dupe them with deceptive advertising.
- We respect staff when we treat them as individual co-workers with valuable experience and insight that we may lack; we don't practice "mushroom management".[4]
- We respect ourselves by advocating strongly for our conclusions, to the degree the data warrants; we don't cave just because somebody in senior leadership doesn't like what we have to say.
- We respect other stakeholders—members of the local community who can be affected by traffic, pollution, noise, or demands upon local infrastructure—when we account for those effects in our decisions.

If we fail to respect any of these groups, we risk losing the knowledge and cooperation our process improvement efforts depend on. Allow me to illustrate very simply. Suppose we run a project where we extract valuable knowledge from our front-line workers and devise a simpler, more efficient process. Then, after implementing the improved process, we lay off everybody who helped us build it. Management gets bonuses; the staff gets screwed.

And then what happens on the next project? Management announces a new process improvement initiative. Consultants fly in. The staff has good reason to withhold what it knows. The next project may still extract a gain, but management has made cooperation harder and left itself less able to learn from the work. Lather, rinse, repeat; forever and ever, amen.

In real process improvement, we value people as individuals with valuable accumulated knowledge—*because that's what they are.*

So, what do you do if your management wants you to run a "process improvement" project built around treating their staff like garbage?

Don't. Refuse to help them disguise exploitation as "process improvement."

### 4.5.2. Continuous Improvement and Ethical Competition

The other pillar of process improvement is continuous improvement. We've already seen one dimension of the philosophy of continuous improvement. We are always seeking to make our processes better. But there is also a deeper, ethical dimension of continuous improvement as well. The philoso-

[4] "Feed 'em bullshit and keep 'em in the dark."

phy of continuous improvement tells firms to compete in the marketplace by becoming better at creating value.[5]

Let's illustrate the importance of this admonition by considering its alternative. Business competition can reward excellence, but high stakes also tempt firms to seek those rewards through deception, coercion, manipulation, or the transfer of costs to people who did not consent to bear them.

For example:

- Concealing known defects or safety risks from customers, workers, or regulators
- Designing products to fail prematurely or making repair unnecessarily difficult
- Using deceptive pricing, hidden fees, subscriptions, or "dark patterns" to undermine informed customer choice
- Exploiting customers' personal data in ways they did not meaningfully understand or consent to
- Manipulating performance measures, financial statements, or quality records to create the appearance of success
- Pressuring employees to work off the clock, falsify records, skip safety procedures, or conceal problems
- Retaliating against employees who report defects, hazards, fraud, or mistreatment
- Treating layoffs, chronic understaffing, and work intensification as the primary means of improving "productivity"
- Transferring unreasonable risks and costs to suppliers while demanding unsustainable prices or delivery schedules
- Deliberately creating dependence on a proprietary platform and then exploiting customers or suppliers once switching becomes costly
- Copying competitors' work through theft, espionage, or deliberate infringement rather than developing better capabilities
- Using market power to suppress wages, dictate unfair supplier terms, or acquire competitors merely to eliminate them
- Privatizing gains while shifting the costs of pollution, illness, infrastructure, or financial failure onto the public
- Meeting the letter of a law while deliberately defeating its purpose through regulatory or tax arbitrage

[5] Liker describes Toyota as welcoming competition because it drives continuous improvement and produces better products for customers (Liker 2021, 47). I take this to imply an ethical claim about how firms should compete: by becoming better at creating value, rather than through deception, coercion, or shifting costs onto others. Likewise, Hennig and Romar (2023, 347) interprets the two pillars as explicitly ethical commitments aligned with a humanistic style of management.

If your "business strategy" relies on any of the above, process improvement cannot make it ethical. That strategy needs to change; I will not help dress it up as improvement.

Continuous improvement offers a different path: we compete by understanding our customers and what they value, innovating to create better offerings, and improving how work is done. Increased efficiency and quality can create room to lower prices, invest in people, and withstand setbacks. Training and retaining a skilled workforce builds capacity for further innovation and improvement.

Incentives matter. A firm that rewards short-term results gets short-term behavior: deferred maintenance, concealed defects, excessive risks, mistreating customers, and the unethical behaviors cataloged above. But each shortcut undermines the business, subtly or not so subtly.

And of course, eventually the business cycle goes bust. When cash is tight and credit expensive, expect your customers to examine the value your firm offers much more critically. If you've built your business on the bedrock of delivering superior value, you can welcome their scrutiny. If you haven't—well, it's always a good idea to keep your résumé up-to-date.

Nothing *guarantees* success in business; sometimes sheer bad luck can wreck a firm, no matter how well-managed. Even so, I commend to you the philosophy of process improvement described in this chapter. It is not only prudent management; it's also the right thing to do.

These two pillars will only show up occasionally as explicit discussion topics below. However, they are almost always present in the background.

The Wash n' Fold case gives these foundations a concrete thread through the rest of White Belt: the laundromat is the firm, the person receiving the clean laundry is the customer, and the observed sequence of work is the process by which that value is created. In the White Belt capstone, these distinctions become the basis for deciding whether a proposed change improves the customer's experience without treating the people doing the work as expendable inputs.

### 4.6. Conclusion

When we improve a process, we are improving the mechanism by which the firm creates value for its customers. This is why process improvement is so crucial. Better processes can improve service and free resources for investment in the business and its people. Whether that happens depends on the costs of the change, market conditions, and what the firm chooses to do with the gains.

### 4.7. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** Understand In your own words, explain how the four foundational concepts — firm, customer, value, and process — are related. Your answer should be a short paragraph that could serve as an explanation to a colleague who has never studied process improvement.

---

**2.** Understand Using the Wash n' Fold case, identify the firm, the customer, the core value the customer receives, and the process being improved. Write your answer in 3–4 sentences.

---

**3.** Understand In your own words, define a **process run** and explain why observing multiple runs is better than observing only one.

---

**4.** Understand Think about a process you participate in regularly — at work, at school, or in daily life. In one or two sentences, describe the process and identify who the customer is. Then: does the firm or organization running that process seem to understand what you, as the customer, actually value? Give one piece of evidence for your view.

---

**5.** Understand In plain language, explain what process variants are and why we care about them for process improvement work.

---

**6.** Understand Why can a process look busy internally while still underperforming for customers? Answer using value and process concepts from this chapter.

---

**7.** (Understand) Write a two-sentence explanation of why process thinking is a prerequisite for meaningful improvement work.

---

**8.** (Understand) Explain why continuous improvement and respect for people must operate together. Give one example of an improvement practice that supports both foundations.

### 4.7.1. Reflection Questions

Write 3–4 sentences about one place where your organization optimizes internal convenience at the expense of customer value, and how you would recognize that pattern earlier next time.

Write three paragraphs about a time when you were asked to do something you didn't agree with at work. In the first paragraph describe the situation, what you were asked to do, and why you didn't agree with it. In the second, describe the actions you took and the result. In the final paragraph reflect on what you might have done differently.

## 4.8. Chapter Summary

- Firms create value for customers through processes made of observable steps.
- The customer is the recipient of the benefit of the firm's services or the use of its goods, regardless of who pays; internal recipients can be treated "as if" they were customers when practical.
- Value is judged from that customer's point of view
- A process run is one complete instance traced from a defined start to a defined stop.
- Process variants may be necessary or wasteful; improvement requires distinguishing intentional paths from unintended ones.
- The Toyota Way pairs continuous improvement with respect for people so that better performance does not come through exploitation.

# 5.

# The Five Principles of Lean

> Everything should be made as simple as possible, but not simpler.
> — Albert Einstein

## 5.1. Why This Chapter Matters

In the last chapter, we learned the basic vocabulary of lean: firm, customer, value and process. But it's not enough to know firms create value through processes. When we're trying to improve a process we still need to know:

- What should we look for?
- What should better process behavior feel like?
- What kinds of system design usually support it?

This chapter introduces five core Lean principles that help answer those questions. With these principles, teams will know what they are looking for when they map and measure work. Think of the principles as telling us what kind of system we're trying to build.

## 5.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Lean Six Sigma Principles and Tools Explain the five Lean principles as a sequence for directing process improvement, from defining customer value through pursuing continued improvement **Understand**
- Lean Six Sigma Principles and Tools Explain the difference between flow and pull by contrasting continuous-flow behavior with batching and pull control with push scheduling **Understand**

- Lean Six Sigma Principles and Tools Analyze a simple process to identify likely violations of Lean principles and justify each diagnosis with observable process behavior **Analyze**

### 5.3. Five Principles Overview

There are many methodologies and frameworks for process improvement. At the White Belt level, we will focus on practical fundamentals of the Lean side of Lean Six Sigma. The history of Lean and the production systems that preceded it will have to wait for later. Here, our concern is practical: what direction should process improvement take?

The heart of the Lean methodology can be summarized in five core principles:

1. Define value from the customer point of view
2. Map the process[1]
3. Create flow
4. Establish pull
5. Pursue perfection

The order matters. First, we identify the outcome the customer values; otherwise, we may efficiently produce the wrong thing. Second, we make the process visible so that we can examine how work actually creates that outcome. Third, we improve the movement of work by reducing unnecessary waiting, interruption, and batching. Fourth, we control when work begins so that actual demand and downstream capacity—not arbitrary schedules—govern its release. Finally, we repeat the cycle because customer needs and operating conditions continue to change.

The principles are easy to state but not always intuitive to apply. Premature local optimizations can obstruct system-wide flow. Keeping every person or machine busy can create more work than the process can absorb. Engineering a better process usually involves thinking carefully about tradeoffs just like this. The five principles give us a direction for judging those tradeoffs.

[1] We take these principles from (Womack and Jones 1996), which names the second principle "identify the value stream." This book substitutes *map the process* at White Belt level to establish the underlying observational and modeling skills before introducing fuller Value Stream Mapping practice at later belt levels.

### 5.4. Defining Value from the Customer Point of View

The previous chapter established that the customer, not the firm, defines value. The practical question is to figure out what the customer values.

At White Belt level, begin by asking. A face-to-face conversation can reveal what outcome the customer seeks, what frustrates them, and what tradeoffs matter most. Surveys, focus groups, complaints, reviews, and interviews provide other ways to listen. In digital services, usage data can also show what customers actually do.

The effort to understand customer needs is often called listening to the *Voice of the Customer*, or *VoC*. You may also encounter the voice of the business, the voice of employees, and the voice of the process. The voice of the business identifies financial, strategic, legal, and operational constraints. Employees contribute knowledge about how the work is actually performed. Finally, in many cases, especially in the world of manufacturing, the process itself produces evidence about its own timing, variation, defects, and capacity in the form of event logs, error messages, dashboards, and so on. All four voices matter, but they answer different questions. The customer's voice remains primary when defining value because the firm exists to deliver an outcome the customer (or somebody acting on his or her behalf) is willing to fund.

Designing rigorous surveys or analyzing behavioral data is beyond the scope of this chapter, but one caution matters now: asking is a starting point, not proof. Customers may struggle to articulate what they need, and what they say may differ from what they choose, use, or reject. Whenever possible, we should compare stated preferences with actual, observable behavior.

Because the Wash n' Fold case study is fictional, there aren't any actual customers to talk to. So, instead just try to imagine what you would like your laundry to be like, if you were a customer of this business.

You probably came up with a short list like:

- clean
- undamaged
- dry
- folded
- available for pickup no later than 5:00 p.m.

We don't need a fancy tool or template to do this exercise, but it is good to bring all of these requirements to mind before proceeding, even if the only thing customers are currently complaining about is the pickup time.

Often it is helpful to think of value in terms of customer specifications.

**Definition 5.1.**

> **Definition Customer Specifications**
>
> A customer specification is a precise, measurable property the finished good or service needs to have in order to meet the customer's need.

Usually we have to do a little work to translate vague customer feedback into true customer specifications. Customers want tires that are long-lasting, safe, provide adequate traction, and so on. It's up to the folks at Michelin to figure out how long, "long-lasting" is and how to measure it. Is it 30,000 miles or 50,000? What does "safe" really entail in terms of stopping distance at 30 mph or 60 mph? Take a step back and ask yourself what are these engineers really doing?

They are trying to understand what the customer values and translate that into a set of specifications they can design the tire to meet.

In our case, the key customer specification is really easy. The laundry just needs to be dry and folded and ready for pickup by 5:00 p.m. At later belt levels, translating customer feedback into explicit specifications becomes less straightforward.

### 5.5. Mapping the Process

Real processes can be distributed across many different people, systems, departments, and sometimes even different organizations. Each process participant usually sees only their own local piece of that vast complex system. A process map creates an end-to-end model so that the team can examine how those pieces interact.

But what exactly are we looking for from the process map?

In the first place we are looking for what has to happen to deliver the quality the customer expects. What does the firm have to do to create laundry that is clean, undamaged, dry, and folded? We'll obviously be looking for a "wash" step, a "dry" step, and a "fold" step. But what our map is going to do is help us see *what else is taking place* in the process outside those obvious steps.

Mapping makes boundaries, sequence, handoffs, queues, decision points, rework, and responsibility visible. By doing so, the map can reveal that an apparent problem in one department is produced by an upstream decision or that improving one step merely moves the queue somewhere else. Strictly speaking, anything we see in our process that isn't creating value for the

customer is waste. In Chapter 10 we will detail eight common kinds of waste and what we do about them. For now, think of waste as anything in the process that isn't needed to deliver value to the customer.

The process map is a very important tool, but remember it is still only a model. It must be checked against observation, process records, and the knowledge of people who perform the work. Mapping the official procedure without observing the actual process can hide the very behavior we need to improve. This is why we always begin by mapping the work *as we observe the staff actually perform it*, not just according to some specification or standard operating procedure document.

> Pro Tip
>
> Managers and supervisors often understand a business process based on official documentation or policy. But policies are often ambiguous and rarely fully specify what exactly has to be done. Different offices or even different employees in the same office may have different interpretations of the policy. This is a common source of process variants. So it often comes as a surprise to managers and supervisors to discover that their supposedly "simple" process is being performed differently by different people.
> Standardizing a process on the most efficient variant is a common solution to process problems in that case. But we must standardize very carefully.
> Why did the employees perform the work in different ways? Were there software limitations that had to be worked around? Are there different local regulations in this location that force changes? All this has to be investigated thoroughly.

At the White Belt level, we begin with a bounded process and a simple mapping method I will call the **Simplest Possible Process Map**, or **SPPM Method**. The SPPM captures the information needed for a first improvement cycle. We will learn to create and validate an SPPM in Section 7.1. Then in Chapter 8 we will build an SPPM for the Wash n' Fold laundry process. More advanced belt levels will introduce additional mapping methods.

### 5.6. Creating Flow

The third principle concerns the movement of work. Once a process run begins, how directly does the unit travel from one step to the next? Does it move continuously, or does it stop in queues, wait for a batch to fill, return for rework, and compete with other priorities?

In our Wash n' Fold case, we might investigate whether loads of laundry are batched and if so how much time specific loads of laundry spend in a queue waiting for the next batch to run. Batch queues are a frequent, but usually invisible time suck in a process.

To see why, notice flow depends on more than just the sequence of steps in the process itself. It depends on how the firm coordinates the people, locates and services equipment, stores and retrieves materials, and communicates information needed to perform those steps. That larger arrangement is called *the production system*.

**Definition 5.2.**

> **Definition: Production System**
>
> A production system is the organization of people, processes, machinery, materials, and information necessary to create a product or provide a service.

Batching and continuous flow are two diametrically opposite ways to organize a production system.

**Definition 5.3.**

> **Definition: Batch Processing Production System**
>
> A batch-processing production system accumulates multiple units and processes them as a group before releasing the group to the next step.

**Definition 5.4.**

> **Definition: Continuous Flow Production System**
>
> In a continuous-flow production system, each unit begins the next step as soon as the preceding step is complete and the next step is available.

Batching can appear efficient. Every time we have to change over from producing one thing to another, we have some overhead. We may have to set up a machine, retrieve tools, clean equipment, or transport material. A larger batch spreads that setup cost across more units. In some cases, batching may even be necessary. Examples include:

- changing the dies on a mechanical press;
- firing multiple ceramic pieces in the same kiln; and
- transporting material when each trip has a substantial fixed cost.

But the local efficiency of a large batch often creates system-level costs. The first completed unit must wait for the rest of its batch before moving forward. Work accumulates between steps, increasing lead time and consuming space. A defect discovered at the next step may already have been repeated across the entire batch. Large batches also make it harder to change priorities or respond quickly when customer demand changes.

For this reason, large quantities of work in progress waiting around for processing are often a red flag for process improvement, as we shall discuss in more detail in Chapter 10.

Continuous flow reduces those delays by moving units forward in smaller quantities. It shortens feedback loops, exposes problems sooner, and makes the current state of the process easier to see. The directional principle is therefore to reduce batch size and interruption as far as practical. Not every process can operate literally one unit at a time, but in general the smaller and more frequent you can make your batches, the healthier and more effective your process will be.

It is helpful to see these principles in action. The following video is optional; the surrounding text contains the instruction needed for the White Belt path. The demonstration uses the term "one-piece flow" for what I'm calling "continuous flow," but it is the same idea:

> **Optional Video: One Piece Flow vs. Batch Processing**
>
> Watch "One Piece Flow vs. Batch Processing" on YouTube (GEMBA Academy, ~5 min)
> This book links to the publisher's copy; it does not reproduce the video.

> **Media Credit**
>
> Source: *One Piece Flow vs. Batch Processing*, YouTube. Publisher and uploader: GEMBA Academy.
> Rights: Reference link only; no recording, still, or transcript is redistributed or relicensed by this book.

**One Piece Flow vs. Batch Processing**

Video available online

https://youtu.be/Dr67i5SdXiM

Use the written activity below when offline.

When examining a process, ask where work stops after it has begun and why. Those stopping points tell us how well the work flows much more accurately than looking to see whether the employees look busy.

In practice, creating flow means removing evidenced obstacles to the next step, preparing handoffs, reducing unnecessary travel and batching, and arranging enough capacity to meet the customer's required pace. Once we have learned to measure that pace and the actual work, Chapter 11 will show how to draft a small flow-and-pull experiment, including a work limit and a release signal.

### 5.7. Establishing Pull

Flow and pull solve different problems. Flow concerns how work moves after it begins. Pull concerns what authorizes work to begin or move forward.

In a push production system, work is released according to a forecast, schedule, quota, or other upstream decision. The upstream step sends work forward whether or not the next step is ready to receive it. Because the downstream step has no mechanism for limiting what the upstream step releases, push systems tend to create excess work in process when demand, staffing, or capacity differs from the plan.

In a pull production system, by contrast, new work begins only in response to a signal from actual demand or a downstream step. That signal might be a customer order, an open appointment slot, or the withdrawal of an item from a controlled buffer. A buffer is a designated, limited place where a small amount of completed work waits for the next step. For example, a rack between washing and folding might hold at most five washed loads. When the folder takes one load from the rack, its empty space authorizes

the washer to send one replacement rather than another uncontrolled batch. The downstream step takes only what it needs, and the upstream step replaces only what was taken.

The pull rule should also limit the amount of work in progress authorized at any one time. One common method uses a fixed number of *kanban*, or work signals, such as cards that travel with the work. If the five-load rack has five cards, all five cards are attached to loads whenever the rack is full. No additional load may begin until the folder takes a load and returns its card. This makes the work limit visible and prevents upstream activity from overwhelming downstream capacity.

Batch-and-push systems often begin with reasonable concerns. A setup may be expensive, a forecast may be the best demand information available, and a fixed batch may appear to protect the next step from interruption. The trouble begins when the release decision and the performance incentives ignore what the customer needs and what downstream work can absorb.

Now imagine that Sam, a factory manager, is rewarded with a cash bonus for meeting a monthly production quota. Unfortunately the forecasted demand simply never materializes. Demand falls short of the forecast, but the plant can still make the planned quantity. Stop and think for a minute: what would you expect Sam to do? Will he reduce production because there is no demand to satisfy, or continue production in order to hit the quota? What will the consequences to the firm be, and where do you expect those consequences would appear?

Well, since Sam is still responsible for the same production target and has a financial incentive to hit it, the most likely outcome is he simply plows ahead with production on schedule and the firm ends up with a warehouse full of stuff it can't sell. The Chief Operations Officer asks Sam, "Why did you keep making a bunch of stuff we can't sell?"

"I'm just doing what you explicitly told me to do," Sam replies. "If you're looking for someone to blame, blame yourself for getting the forecast wrong!"

The instruction and the reward still favor production even after the demand that justified the target has disappeared. People can make individually defensible decisions inside a system that rewards output while exporting delay, rework, risk, stress, and exhaustion to everyone involved.

Flow and pull change the whole operating logic. Work enters in response to actual customer demand, downstream consumption, or available capacity, and a visible work-in-process limit prevents the upstream step from releasing more merely to stay busy. Smaller transfers allow the next step to begin sooner. They also shorten the feedback loop: a defect found on the first unit can be corrected before it is repeated across a large batch.

Pull does not magically make variability disappear. A late supplier can still stop production, a storm can still close the plant, and a staffing loss can still reduce capacity. A queue of customer demand may grow outside the controlled process, and a work limit does not create the missing capacity needed to serve it. What pull does is keep the system from *disguising* those problems as productive activity. The shortage, blockage, or capacity gap becomes visible, which gives management a chance to address the actual constraint in real time instead of burying it under more work in process.

For now, we can state the expected benefits as hypotheses to test:

- Shorter delivery time when work no longer waits unnecessarily for a batch.
- Earlier detection of quality problems because feedback travels across smaller transfers.
- Lower inventory, storage, and rework costs when the process releases only what demand and capacity authorize.
- Less discounting and obsolescence when the firm avoids producing unsold finished goods from an inaccurate forecast.
- More visible overload because a work-in-process limit prevents the active queue from growing without bound.

These benefits depend on the process, its constraints, and how faithfully the method is carried out. The comparison identifies a mechanism worth testing while leaving room for batches and forecasts where the process requires them.

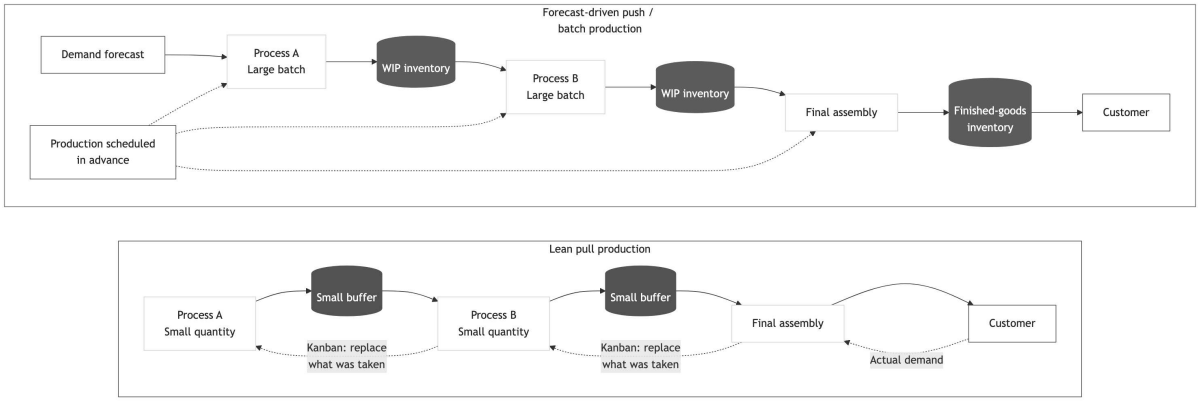



**Figure 5.1.:** Forecast-driven mass production compared with kanban-controlled pull production.

### 5.8. Pursue Perfection

The fifth principle does not claim that a literally perfect process is attainable. Rather, the principle simply says the current process is never final. Customer expectations change, new technology becomes available, competitors improve, and yesterday's solution creates tomorrow's constraints. Improvement must therefore be continuous.

Pursuing perfection sends us back through the preceding four principles. Has the customer's definition of value changed? Does the map still describe the work as it is actually performed? Where does work now wait or return for correction? Are pull signals still aligned with demand and capacity? Each improvement changes the process, so each improvement creates a new current state to observe and question.

This does not mean trying to fix everything at once. Most improvement proceeds through many bounded experiments whose gains accumulate over time. A team identifies a problem, tests a countermeasure, studies the result, and either adopts, revises, or abandons the change. The next chapter introduces a structured cycle for doing that work.

But incremental improvement also has limits. If a process no longer delivers any meaningful value, or if its basic design *cannot* meet customer needs, tuning the existing steps may be the wrong response. You don't need to worry about tuning your brakes and shocks for maximum performance if the car is up on blocks and the transmission is sitting on the pavement. The firm may need to redesign the process rather than improve it incrementally. Rebuilding a process from the ground up is known as Business Process Reengineering and Lean Six Sigma provides tools to accomplish that work as well. But those tools are out of scope at the White Belt level.

At this level, the important direction is: improve a functioning process toward a better state, but do not use continuous improvement as an excuse to preserve a process whose basic purpose or design has failed.

### 5.9. Five Principles, Two Pillars

The five principles tell us what direction improvement should take. The two pillars introduced in Section 4.5 constrain how we pursue that direction. Continuous improvement keeps the principles active over time; respect for people keeps improvement from becoming a pretext for deception, coercion, or exploitation.

| Five Lean principles | Continuous improvement | Respect for people |
|---|---|---|
| Define value | Continually test what customers truly need | Respect customers by defining value from their perspective |
| Map the process | Expose waste and improvement opportunities | Involve the people who actually perform the work |
| Create flow | Experiment to remove interruptions and delays | Design work that is safe, understandable, and sustainable |
| Establish pull | Use demand signals to improve coordination | Avoid burdening people with overproduction and chaotic priorities |
| Pursue perfection | Return to the first four after a change | Develop people as capable problem-solvers rather than treating them as interchangeable labor |

The practical connection between the principles and the pillars is trust. People who perform the work often know where the delays, workarounds, and recurring failures are hidden. The organization needs that knowledge to define the current state and improve it. If management uses employee knowledge against the people who provide it, future improvement efforts will lose access to the truth. Chapter 4 examined that problem in more detail; here, the point is simply that the five principles cannot be separated from the conditions under which people are asked to apply them.

This dependence on trust also helps explain why transforming a batch-and-push operation is so difficult. Lean changes more than a firm's tools and terminology; it changes what the firm measures, authorizes, and rewards. A Fortune 500 company cannot merely announce an improvement initiative and expect the whole enterprise to become Lean three months later. Leaders must change how they respond when workers expose problems—and, as the example of Sam the factory manager showed, they may need to replace incentives that reward production no customer needs. The upshot is that an organization is not Lean merely because it uses *kanban* cards and moves work in small batches. Trust between staff and management and a shared commitment to delivering quality to the customer are indispensable. *Kanban* cannot repair a culture that punishes candor or rewards unwanted production.

## 5.10. Conclusion

The five principles describe the direction of Lean improvement. Begin with the outcome the customer values. Make the process visible. Improve how work moves through it. Control the release of work through signals tied to actual customer demand and capacity. Then return to the beginning and ask what should improve next.

These principles do not prescribe one ideal process for every setting. They give us a disciplined way to question the current process and judge proposed changes. The Wash n' Fold case gives us a continuing example, and we will use this language throughout the White Belt capstone to explain why a problem matters and what better process behavior should look like. The next chapter introduces PDCA, the structured cycle we will use to put this direction into practice.

## 5.11. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Understand) Explain why the five Lean principles are presented in this order: define value, map the process, create flow, establish pull, and pursue perfection. For each transition, state why the next principle is needed.

---

**2.** (Understand) Classify the flow and release logic in each process.

- A bakery decides each morning to produce 200 loaves from its sales forecast. Once production begins, each loaf moves from shaping to proofing to baking with little waiting.
- A custom-print shop begins a job only after receiving a customer order, but then holds that job until 50 similar orders accumulate for printing.

For each process, state whether it demonstrates relatively good or poor flow and whether its release is push or pull.

---

**3.** (Analyze) A hospital receives referrals throughout the day. Intake staff enter each referral immediately, but at 4 p.m. the system releases every accumulated referral to the scheduling queue, whether or not the scheduling team has capacity. Schedulers begin contacting patients the following morning.

Identify one flow problem and one pull-control problem. State what evidence you would collect before recommending a change.

---

**4.** (Analyze) Apply all five Lean principles to the Wash n' Fold case. For each principle, write one diagnostic question and identify one piece of evidence you would seek before proposing a countermeasure.

---

**5.** (Apply) A team has reduced the average approval time in a functioning service process from five days to two. The service now meets most customer needs, but 10 percent of requests still take more than a week. One manager declares the process fixed; another proposes replacing the entire service immediately.

Using **Pursue Perfection**, recommend a more disciplined next step.

### 5.11.1. Journal and Reflect

Write a short paragraph about one place in your own work where one of the five Lean principles challenges or even changes how you think about that process. State which principle it is, what current behavior it makes you question, and what kind of evidence you would want next to test a change.

Imagine that a proposed flow or pull change would improve speed but might transfer more interruption, uncertainty, or physical burden to another person. What would you ask that person, and what guardrail would you include before testing the change?

## 5.12. Chapter Summary

- The five Lean principles form a sequence: define customer value, map the process, create flow, establish pull, and pursue perfection.
- A process map is a model of the actual work and must be checked against observation, records, and the knowledge of the people who perform it.
- Flow concerns how work moves after it begins; pull concerns what authorizes work to begin or move forward.
- Smaller batches and pull controls can reduce waiting and excess work in process, but they are directional design principles rather than absolute rules for every process.

- Pursuing perfection means repeatedly learning and improving within the constraints of continuous improvement and respect for people; it does not mean that literal perfection is attainable.

# 6.

# A Structured Approach to Process Improvement

> "A problem well-stated is a problem half-solved." — Charles Kettering
>
> "By failing to prepare, you are preparing to fail." — Benjamin Franklin

## 6.1. Why This Chapter Matters

Imagine your team has late deliveries, frustrated customers, and three competing opinions about the cause. Do you hire more people, buy new equipment, or redesign the workflow? This chapter introduces a very simple management tool, called the PDCA cycle, to guide our improvement work so we start with structured reasoning rather than guesswork. It also helps the team achieve a shared rhythm and report meaningfully to stakeholders.

## 6.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Lean Six Sigma Principles and Tools Explain the PDCA cycle and how each phase connects to the next. **Understand**
- Leading and Managing Projects Distinguish the work of the Plan, Do, Check, and Act phases in a simple improvement project. **Understand**
- Leading and Managing Projects Write a clear, factual problem statement tied to customer impact. **Apply**

- Leading and Managing Projects Formulate a practical SMART goal for an improvement cycle. **Apply**

### 6.3. The PDCA Cycle

In the last chapter we learned a set of Lean principles: define value from the customer point of view, map the process, create flow, establish pull and pursue perfection. But how do we go about actually applying these principles to perform process improvement?

A haphazard approach—guessing what is wrong and trying random fixes—is usually not reliable.

- Which fixes should we try first?
- Why do we think those are going to work?
- How would we know if our fixes are helping or hurting?
- How would we know one fix would not interfere with another?

These are common difficulties when process improvement is unstructured.

Instead, what we need is a structured way of implementing process improvement. Later, we will learn there are multiple frameworks we can use to structure process-improvement projects. For our purposes in this chapter, we will use the simplest and most practical framework that fits this problem. This framework is called PDCA and it is a four-phase, repeatable cycle:

1. *Plan*
2. *Do*
3. *Check*
4. *Act*

**Figure 6.1.:** Deming cycle (PDCA) diagram from Wikimedia Commons

**Media Credit**

Source: File:PDCA-Loop.png, Wikimedia Commons. Creator: Christoph Roser at AllAboutLean.com.
Rights: CC BY-SA 4.0. Used under license (not fair use).
Modification: No substantive modification; resized by the publication renderer.

Note

W. Edwards Deming popularized this cycle in modern quality management practice, attributing the original to Walter Shewhart. In some books and organizations, you will also see it written as **PDSA** (Plan-Do-Study-Act). In this chapter, we use the PDCA label.

The cycle matters for more than orderly project management. In this book, reducing variation and eliminating waste also serve a further aim: helping an organization learn from the results of its decisions. Waste and variation give us important things to investigate, while PDCA supplies the feedback loop: make an explicit plan, test it, compare the result with what you expected, and decide what the evidence warrants next. The White Belt chapters follow that loop once from beginning to end so that each map, measure, and countermeasure becomes part of one connected argument rather than an isolated tool exercise.

PDCA is the right-sized framework for that first bounded cycle. At Yellow Belt, we will organize a more substantial improvement story with A3; later belts will use DMAIC when the problem requires deeper measurement, analysis, and project discipline. These frameworks differ in scale and emphasis, but they share the same obligation to learn from evidence before standardizing a change.

The PDCA cycle is a good tool for managing simple projects, where the problem we are investigating is confined to a single business unit with a specific, measurable problem. Our fictional Wash n' Fold case is a good candidate. Over the next several chapters, we run this cycle using the Wash n' Fold process as a concrete case. By working through this case together, we see the Lean principles in action and practice simple but powerful tools. Let's begin with the Plan phase.

## 6.4. Wash n' Fold Project: Plan

We need to do three things as part of the 'Plan' phase.

1. Define the problem and set a clear goal.
2. Map the current process.
3. Establish baseline metrics.

By defining the problem and goal, we help narrow our scope. Mapping the current process gives us understanding of the process and its variants. The map also makes it easy to calculate a number of convenient metrics. This is useful for two reasons. First, viewing the map and its metrics can help identify potential interventions to try in the "Do" phase. Second, these metrics form the baseline to judge the effectiveness of these interventions in the "Check" phase.

For this project, we follow one customer order from drop-off until the completed order is staged and the customer is notified it is ready. Customer travel, pickup, and payment are outside our scope for this project.

### 6.4.1. Define the problem

The Plan phase starts with a clear, neutral problem statement. What we want is a specific, observable claim about what is happening. Not *why* the problem is happening or what we should do about it. At this step, we are *only* defining the problem, not attempting to solve it or assign blame. Do not assume a cause and do not propose a fix yet.

Try to tie the problem statement to its impact on the customer. “Customer orders are late” is a good start on a problem statement. “Employees are not working fast enough,” is a bad problem statement because it jumps to blame and an unverified cause. “We need another washer” isn’t a problem statement at all; it’s just jumping to a potential solution based on an unverified cause.

If you find yourself struggling with your problem statement, it can be helpful to write these questions on a piece of paper:

- Who is being affected?
- What is the problem affecting them?
- When is the problem happening?
- Where is the problem happening?
- How is the problem affecting the customer?

Use these five questions as a lens to think through what you’ve observed systematically. Are there specific subgroups of customers or employees being affected or is it everybody? What exactly is happening? Is there a specific time or place where it occurs more frequently than elsewhere? Is the problem confined to a specific area or location? Thinking through such questions can help you make your problem statements more specific and detailed. Avoid the temptation at this point to start speculating about the “Why?” Absolutely avoid assigning blame. Note too what newly discovered information might cause you to revisit and revise your problem statement.

> Pro Tip: Trust the Process
>
> I am intentionally belaboring what might seem like a simple point because, in practice, writing a good problem statement is often one of the biggest challenges process improvement beginners face. For some reason—psychological, cultural, or otherwise—our natural tendency is to jump straightaway into analyzing and solving problems.
> Experienced mentors and facilitators often need to remind the project team we must proceed methodically to get a good result. You may find it helpful to remind yourself and your team to “trust the process”: gather evidence before settling on a solution. That discipline includes revising or rejecting our preferred explanation when the evidence contradicts it. We will get to solutions after we’ve gathered and analyzed data to guide our problem solving.

**Figure 6.2.:** Trust the Process

> ❗ Quick Write: Define the Problem
>
> Go back to the Case Study and write a short problem statement (3–5 sentences).
> Focus on what is happening now, who is affected, and why it matters.
> Do not diagnose the problem or propose solutions yet.

> 💡 Sample Answer
>
> Your answer should look something like:
> "Recent Friday evidence shows approximately 28% of same-day orders dropped off by 9:00 a.m. were not staged and ready, with the customer notified, by the 5:00 p.m. promise. (What and When) Customers relying on that same-day promise cannot consistently tell when their laundry is ready. (Who and How) This performance gap weakens the service's promise of convenient, reliable same-day laundry. (How—Business)"

### 6.4.2. Set a SMART Goal

Once we have stated the problem, our next task in the Plan phase is to define what improvement should look like in this cycle. At White Belt level, a practical way to do this is a SMART goal. If you're not familiar with the acronym, it summarizes five properties a good goal ought to have.

- *Specific:* states exactly what should improve.
- *Measurable:* uses at least one observable metric.
- *Achievable:* realistic for the team and timeframe.
- *Relevant:* tied to customer impact and business need.
- *Time-bound:* includes a clear deadline.

For this case, a **same-day promise violation** means an order dropped off by 9:00 a.m. that is not staged and ready, with the customer notified, by 5:00 p.m. "Reduce same-day promise violations from approximately 28% to 10% within the next 30 days" gives us a concrete target to shoot for. "Make stuff better" or "improve the process" doesn't.

It's okay if you aren't completely clear at the outset what specific measure your project should target for improvement: we will discuss process metrics more fully in Chapter 9. In practice, it's quite common, at the outset of a project, for the goals not to be completely clear. Often the project goals have to be refined over the course of the project in collaboration with the business owner or manager. The business leadership and the project team need to agree on the target. Without that agreement, the team cannot tell whether its result meets the owner's requirement.

The goal statement should describe the desired future state of the process in as much specific detail as possible. Again, don't include analysis or solutions in your goal statements. Just describe what the world *should* be like if you solved the problem identified in your problem statement.

> ! Quick Write: Write a SMART GOAL
>
> Go back to the Case Study and write a short goal statement (1–2 sentences).
> What does "good" look like?

With the problem defined and the goal clarified, the next step is to see the process clearly enough to act on it. That mapping work belongs to the next chapter. For now let's work some exercises to solidify our understanding.

## 6.5. Conclusion

In Section 3.6, we described five deliverables you'll need to complete the White Belt capstone.

You should now have the material you need to create the first deliverable, which is a small document with your problem statement and goal statement. I encourage you to create a small file on your computer, or write these out officially on a page of your journal for future reference. Together the two statements give us a snapshot of where we currently are, and where we want to be.

## 6.6. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Apply) A small dental practice reviewed 40 scheduled appointments from the previous month. In 18 of them, the patient waited more than 15 minutes past the scheduled appointment time before the hygienist began the cleaning.

Write a one-sentence problem statement for this practice. Then use your imagination to write a weak version of the same statement that violates the "no causes, no blame" rule — and briefly explain what makes it weak.

---

**2.** (Understand) A team is running a small improvement effort on late customer orders. For each activity below, identify whether it belongs primarily to **Plan**, **Do**, **Check**, or **Act**:

- Compare on-time completion before and after the test.
- Write a neutral problem statement tied to missed pickup times.
- Update the standard work and train staff on the new routine.
- Test a new folding schedule for one week.

---

**3.** (Apply) Consider this goal statement: "Improve customer satisfaction soon by making operations better."

Identify at least three SMART criteria this statement fails to meet and rewrite it as a one-sentence SMART goal. For the rewrite, use the Wash n' Fold case's current promise-violation rate of approximately 28%, target of 10%, and 30-day improvement period.

---

**4.** (Apply) A manager wants to skip the Check phase because "everyone can already tell the process feels better." Would you accept that? Why or why not?

---

**5.** (Apply) A university's financial aid office processes student grant applications. Students currently wait an average of 18 business days for a decision; the office's own service standard is 10 days. The director wants to run a PDCA cycle to close that gap.

Write a one-sentence problem statement and a one-sentence SMART goal for this cycle. Keep the problem statement factual and customer-focused; make the goal specific, measurable, and time-bound.

---

**6.** (Understand) What is the difference between a *problem statement* and a *SMART goal*? A teammate drafts the following and labels it as the problem statement for their PDCA cycle:

> "We need to reduce order lead time by 20% over the next month."

Explain what is wrong with calling this a problem statement, and identify the current-state facts you would need before you could write a proper problem statement.

---

**7.** (Apply) Records show that 24% of customer issues handled last month required a callback, while the service standard allows no more than 10%. A supervisor describes the problem this way: "Our team is careless, so customer callbacks are too high." Rewrite the supervisor's statement as a neutral problem statement using the supplied evidence.

---

**8.** (Understand) In one sentence each, define Plan, Do, Check, and Act as used in this chapter.

---

**9.** (Apply) Draft one SMART goal for a process with current on-time completion of 62%, a desired level of 85%, and a 60-day improvement period.

---

**10.** (Apply) A team has one proposed countermeasure but no baseline metric. Write two sentences describing what must be added before entering Do.

---

**11.** (Apply) Write a short Check-phase evaluation note template with three fields your team must fill in.

---

### 6.6.1. Reflection Questions

- Think of an unsuccessful project from work or daily life you've participated in or witnessed. Could that project's failure trace back to having an incorrect understanding of the nature of the problem to be solved or the goal to be achieved? If so, write three or four sentences explaining what you or your team could have done differently.
- Write three or four sentences on which PDCA phase you are most likely to under-resource in real work and what guardrail you will use to prevent that.

## 6.7. Chapter Summary

- PDCA gives improvement work a repeatable rhythm: define the problem and goal, test countermeasures, compare results, and standardize or revise.
- Good Plan phase work is factual and disciplined: it states the problem without blame, it ties that problem to customer impact, and it sets a measurable goal for this cycle.
- The Wash n' Fold case shows why structure matters: busy operations generate many opinions, but a shared method makes decisions easier to test and defend.

# 7.

# Following the Work: Building the Event Log

> "Go see, ask why, show respect." — Fujio Cho
>
> "You can observe a lot by watching." — Yogi Berra

### 7.1. Why This Chapter Matters

"There has to be a better way to do this" is usually true, but intuition alone rarely tells you where to start. A process map is a great tool to help visualize a process, surface problems, and zero in on potential problem areas for further analysis. But if your map is vague or built from memory, the rest of your analysis will be weak. This chapter shows you how to capture the current state of your process with enough discipline that delays, handoffs, and rework become visible. A map drawn from unsupported guesses gives the team a business-themed fairy tale to analyze.

### 7.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Lean Six Sigma Principles and Tools Define and justify a clear start and stop boundary for a process map. **Apply**
- Data Understanding Prepare an event log and use it to preserve individual activity records before aggregation. **Apply**
- Lean Six Sigma Principles and Tools Apply the first five steps of the nine-step current-state mapping workflow with consistent data capture. **Apply**

- Data Understanding Identify queue points, handoff delays, and likely rework loops from observation notes. **Analyze**

In the previous chapter, we began the Plan phase of the PDCA cycle by defining the problem and stating a SMART goal. Together, we said, these provide a view of where the process currently is and where it should be. But how are we going to get there from here? This chapter starts us on our journey towards completing the second vital deliverable of the Plan phase: the production of a process map.

### 7.3. Introduction to Process Mapping

We know the problem at Wash n' Fold: too many same-day orders miss the 5:00 p.m. promise. Suppose we bring the staff together and ask what to do about it. Someone proposes buying another washer. Someone else thinks the employees need to work faster. A third person wants another folding station, while a fourth insists that the real problem is waiting too long to begin washing orders.

Any of these proposals might be right. At this point, however, the problem statement and SMART goal give us no reason to prefer one over the others. They tell us about the result of the process, but they do not tell us how the process produces that result. Jumping directly from a late order to a favorite solution would be a guess, even if it happened to be a lucky one.

What would we need to know before choosing an intervention? We would need to follow an order from drop-off to notification and ask:

1. What happens to the order, and in what sequence?
2. Where is someone actually working on it, and where is it merely waiting?
3. When does responsibility pass from one person or piece of equipment to another?
4. Do all orders travel the same path?
5. When an order takes a different path or circles back for correction, what happened?

Notice what these questions have led us to construct. We have chosen something to follow, given its journey a beginning and an end, placed its activities in order, and marked the waits, handoffs, and alternate paths that might explain the result we care about. Arrange that account so the whole journey can be inspected at once, and what we will have created is what process improvement professionals call a **process map**.

A process map is an end-to-end visual model of what a unit of work actually experiences as it moves through a process. The visual form matters because

the problem may not belong to any one activity viewed by itself. A delay at folding might begin with an earlier batching decision; making one step faster might merely move the queue to the next step. The map holds those relationships together so that people who ordinarily see only their own part of the work can examine the same process.

The map does not, by itself, prove a root cause. It shows us where to ask better questions and which proposed explanations deserve further investigation. It also gives later metrics a structure. Average lead time tells us how long an order took; the map helps us ask where that time accumulated, how much was work, how much was waiting, and which path the order followed. When we return in the Check phase to decide whether a tested change actually improved the process, those distinctions will matter.

There are many styles of process map, each designed to emphasize different features. For our present purposes at White Belt level, the map does not need to be fancy. A hand-drawn flowchart with waiting points, handoffs, and rough times is usually enough to reveal where the investigation should go next.

Three artifacts support the process map, and we should build them in order because each has a different job. First preserve what happened to individual work units in an **event log**. Then calculate the durations recorded in each row and group comparable activity records onto a **mapping worksheet**. The worksheet is not a second place to record the same observations; it is where we *aggregate* the data from the log into summary figures like the count of observations, the average times, and so on. Finally, we use that worksheet to draw the **process map** from the data captured in the log and worksheet.

Now we can make the method explicit. To make your process map, use this nine-step procedure:

1. Define the process boundaries—the start event, stop event, and unit of analysis—and record them in the **event log**.
2. Prepare the **event log** and establish its clock and timing conventions.
3. In the **event log**, follow work units and identify the activities and variants they actually encounter.
4. In the **event log**, record the role responsible for each activity record and each handoff.
5. In the **event log**, record timestamps and factual notes about queues, rework, and interruptions.
6. On the **timing calculation worksheet**, calculate each activity record's processing time (P/T), wait time (W/T), and setup time (S/T) from its recorded timestamps.

7. From the calculation worksheet, build the **map worksheet** by grouping comparable activity records and recording counts, average durations, and value classifications.
8. From the worksheet, draw the **current-state process map**, selecting the queues and branches that need to be visible.
9. Verify the **process map** with a process participant and trace any corrections back to the event log and worksheets.

Steps 3–5 are different things to notice during the same observation, not three separate visits. Do not wait until a work unit is finished to reconstruct its timestamps from memory. The important separation is between recording individual evidence and summarizing it. If the map reveals a gap, return to the observation or participant; do not fill it with an invented average.

The nine steps give us the complete route through the mapping work. This chapter follows Wash n' Fold through Steps 1–5 and produces the event log. Chapter 8 begins by transferring that timestamped evidence to a timing calculation worksheet, then carries Steps 6–9 through the map worksheet, visual map, and participant verification.

Keep the mapping and data collection file guide beside you as you work through Chapters 7–9. It identifies the blank sheets, short worked examples, and full Wash n' Fold references in the downloads. Choose PDF for paper, XLSX for editable worksheets with formulas, or CSV for a portable table without formulas or spreadsheet formatting.

### 7.4. Wash n' Fold Project: Follow the Orders

The case places us in a dedicated Wash n' Fold cell inside a neighborhood laundromat. The cell has two workers, six washer positions, six dryer positions, and two folding stations. The simulation supplies the record of a Friday from opening onward, and we follow customer orders rather than individual garments.

> **i** About the Case Evidence
>
> Wash n' Fold is a constructed case, and its records come from a simulation designed to resemble a plausible Friday in a small laundry. They give us consistent observations to examine without pretending that we visited an actual shop. The same underlying demand and service conditions are preserved for the later current-versus-future comparisons.

The project goal remains the one established in Plan: reduce the percentage of same-day orders dropped off by 9:00 a.m. that are not ready and noti-

fied by 5:00 p.m. from approximately 28% to 10% within 30 days. In the representative Friday used here, 6 of 21 eligible orders missed that promise (28.6%), so our observation begins from evidence consistent with the stated baseline.

### 7.5. Defining the Process Boundaries

Process mapping is an exercise in observation. But before we begin observing things, we need to make some fundamental choices about what exactly we are trying to observe. Specifically, we need to lock down three choices that all have to do with boundaries:

1. *Start event:* when does the customer's clock begin?
2. *Stop event:* when is the customer-facing promise fulfilled?
3. *Unit of analysis:* what single unit are you following (one order, one patient visit, one ticket)?

If these are poorly defined, every metric downstream becomes hard to interpret or even meaningless.

**Definition 7.1.**

> **Definition: Unit of Analysis**
>
> The unit of analysis is the primary "thing" whose passage through the process counts as one run of the process. It might be one customer order, patient visit, service request, batch, or invoice. Choose it before recording observations so that process-level measures refer to comparable cases.

For the Wash n' Fold case, the start and stop events are fairly clear. The process begins when one customer order is dropped off. It ends when the completed order is staged and a readiness text is sent to the phone number recorded at drop-off. Customer travel, pickup, and payment are outside this improvement boundary.

But it's not quite as clear what we should choose as the unit of analysis for this case. Very often, the customer drops off a large load of laundry that won't all fit in one washer or dryer. Further, to keep clothes in their best condition, we need to separate lights from darks and process them in slightly different ways.[1]

[1]Yes, really. See Tide's guide to sorting laundry.

So what should we choose as the unit of analysis in this case? On the one hand, tracking the "customer order" seems very logical. It's the whole order that's dropped off and picked up at the end, after all. On the other hand, one order can get split into two or more loads.

This complex structure is very common to see in process mapping work and it can confound you if you don't sit and think it through carefully. One customer order might involve picking and packing several different items from a warehouse. One patient might require services from several different providers at the same hospital. One service request ticket might need to pass through several different departments, each of which receives its own sub-ticket to work.

In these kinds of real-life cases, I recommend conceptualizing them as having a "one parent, many children" structure. We make the top-level entity (the customer order, the patient visit, the original customer request) the thing we track primarily, and then we capture additional information about the children as needed. So, for our Wash n' Fold case, let's make the unit of analysis the parent **customer order**, which might have one or more child **loads** which are processed separately, whether in parallel or in series. The Order ID therefore remains the case identifier from drop-off through notification. Orders are never mixed, even when sorting divides one order into more than one load. Those loads receive their own Load IDs, which remain linked to the parent Order ID, and reunite before folding.

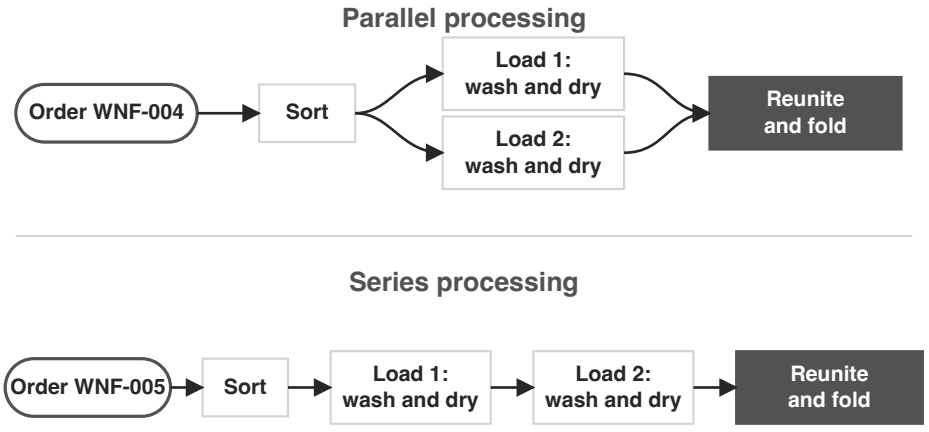



**Figure 7.1.:** One parent order can split into child loads that run at the same time or one after another, then reunite for folding.

Why is the parent/child architecture the right choice for our unit of analysis in this case? Because it keeps the analysis of lead time and promise per-

formance at the level the customer experiences, which is the order. That remains true even though the Wash and Dry steps may need to be recorded at the load level. If that distinction still feels a little tangled, keep reading. It becomes much easier to see once we put it to work.

### 7.6. Prepare the Event Log

We have chosen the order as the case the customer experiences. But Wash and Dry happen to loads, not to the undivided order. How can one log preserve both the customer case and the particular physical item that received the work?

Short answer: it needs a parent case identifier on every row and a child item identifier whenever the work has split.

**Definition 7.2.**

> **Definition: Event Log**
>
> An event log is a row-by-row record of what happens to identified units of work as they move through a process. Each row preserves enough context to say what happened, to which unit, and when. The log is the evidence from which the map worksheet, process map, and order-level summaries are derived.

For this case, the completed Wash n' Fold log is supplied as evidence for us to inspect. The point isn't to reconstruct it from the simulation. Rather, we want to watch each row acquire the information that makes it useful.

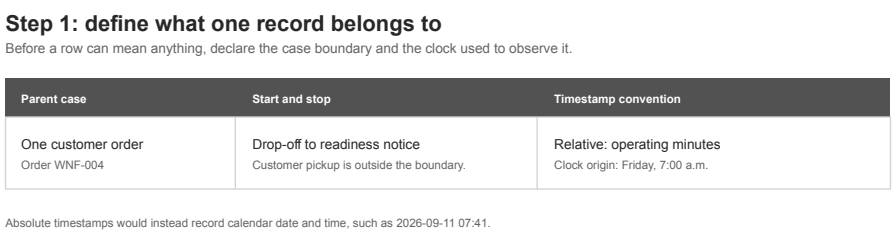



**Figure 7.2.:** Step 1 begins the event log by defining the parent case, process boundary, and timestamp convention.

Before recording a timestamp, choose between **absolute** and **relative** time. An absolute timestamp records a date and time, such as `2026-09-11 07:41`. Use it when work may cross shifts, days, locations, or systems and

another reader must know when it occurred in calendar time. Relative time records elapsed time from a stated clock origin, such as `40.6 operating minutes from Friday at 7:00 a.m.`. It makes intervals easy to compare within one bounded observation window.

An **operating-time clock** counts only periods when the process is open or scheduled to work, so it pauses when the business is closed nights and weekends. A **calendar-time clock** continues to tick during those periods. Either clock can answer a useful question, but they produce different lead times. Name the timestamp convention, clock, time zone when applicable, and clock origin for relative time before comparing records.

**Step 2: preserve the parent case and the child item**

The customer experiences one order, even when a machine acts on one load within that order.

| Parent Case ID | Child Item ID | Why both fields remain |
|---|---|---|
| WNF-004 | L1 | The row stays linked to the order, but names the physical load timed. |

**Order-level activities leave Child Item ID blank; machine activities identify the load.**

**Figure 7.3.:** Step 2 adds the two identifiers needed when an activity acts on a child item within a parent case.

The Parent Case ID remains on every row because the order is the unit of analysis. The Child Item ID is blank for activities such as Drop-off and ticket or Fold and package, which act on the whole order. For Wash and Dry, it identifies each physical load while retaining its link to the order. If work returns to an activity, add another row and increase the pass number.

**Steps 3–5: record what happened, to which unit, and when**

These entries are observations. They are not averages and they do not yet explain why the queue formed.

| Parent ID | Child ID | Activity / pass | Resource | Queue entry | Setup start | Work start | Work end | Factual notes |
|---|---|---|---|---|---|---|---|---|
| WNF-004 | L1 | Wash<br>Pass 1 | washer | 40.6 | 131.4 | 133.0 | 162.4 | Two compatible loads<br>No rework observed |

Use the same timestamp columns for absolute date/time values or relative elapsed time; declare the convention above the log.

Keep participant explanations separate from factual notes, and record a new row when work repeats.

**Figure 7.4.:** Steps 3 through 5 add the activity, resource, timestamps, and factual notes to the same Wash n' Fold record.

The four timestamps are observations. The notes preserve directly observed context, not a conclusion about why the work waited. This row reports that two compatible loads existed and that no rework was observed; it does not claim a cause for the queue.

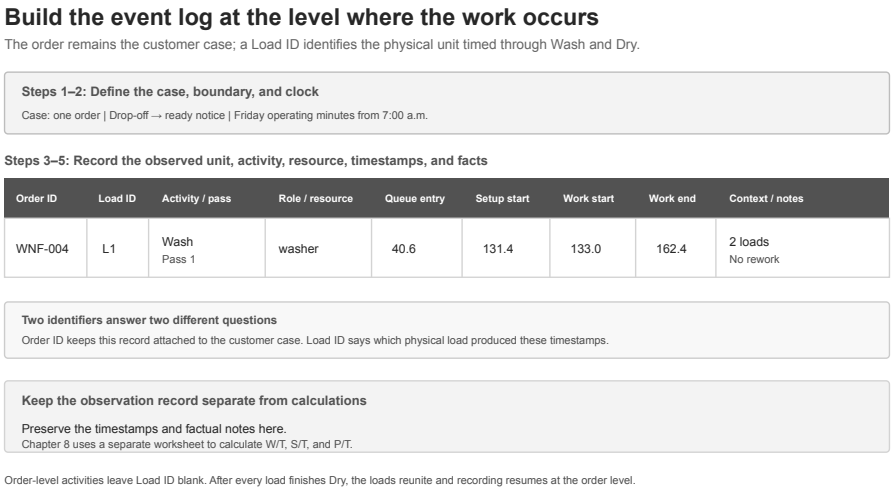



**Figure 7.5.:** Step 5 completes the observation record with a rework mark and factual notes, without turning the record into an analysis worksheet.

The complete record is available in four forms: open the completed Wash n' Fold record in a browser, download the completed PDF, download the completed CSV, or download the completed workbook. These are the full 291-record references. For your first calculation in Chapter 8, use the five-record source excerpt (CSV) or its printable copy (PDF): the `L1` Wash records for `WNF-001` through `WNF-005`. The smaller file preserves the same fields and values as the full log; it selects the records rather than inventing a simpler dataset. These are supplied instructional evidence for this case. Chapter 8 begins by calculating durations from this evidence before grouping or summarizing it.

## 7.7. Follow Work Units Through Major Activities and Variants

The most common mapping mistake is probably trying to map from memory. Memory is simply unreliable here, though. We tend to forget or gloss over small details, and our memories are often shaped more by exceptions than regularities. Since what is abnormal stands out more clearly in memory, describing a process from memory always comes with an innate risk of painting a picture distorted by extremes.

That is why current-state mapping should start with direct observation wherever possible. For White Belt work, your goal is not to document every tiny detail the way a full technical design document would. Rather, your goal is to visualize the real flow (or lack of it) so a team can spot delays, handoffs,

and rework and take action quickly. In other words, we need to make the flow of work and delay visible enough to support the first PDCA cycle.

> Pro Tip: Map Reality, Not Procedure
>
> Document what happens in sequence, including waiting, interruptions, and workarounds. Do not map the policy manual. If procedure and reality differ, that difference is itself useful process data.

This is Step 3 of the workflow: follow work units and name the activities and variants they actually encounter. List the major activities in the order they occur. Again, capture what *does* happen, not what *should* happen.

For first-pass White Belt mapping, target 5–10 major steps. Fewer than five often hide useful detail; more than ten usually means you are mixing levels of detail.

Consider a quick illustration. A team is trying to map an order process:

- **Too broad (3 steps):** Receive order -> Process order -> Deliver order.
- **Usable (7 steps):** Receive -> Verify -> Queue -> Process -> Inspect -> Stage -> Deliver.
- **Too granular (18+ steps):** every click, every form field, every hand motion.

That gives us the right level of detail for the Wash n' Fold process spine:

Drop-off and ticket → Sort and tag → Wash → Dry → Fold and package → Stage and notify.

This sequence names the major activities; it does not mean that every activity acts on the same physical object. The Order ID follows the customer order across the entire process. After Sort and tag, an order may become one or more identified loads. Record Wash and Dry separately for each Load ID, retain the parent Order ID on those rows, and resume order-level recording once all of the order's loads reunite for folding.

### 7.7.1. Process Variants

**Definition 7.3.**

> **Definition: Process Variant**
>
> A process variant is a recurring alternative path for the same process. By "the same process" here we mean that the variants share the same start event, stop event, and primary unit of analysis. They may have different numbers of steps, put the steps in a different order, or differ meaningfully in who performs the steps, the materials involved, or some other important condition.

Process variants are very common in real operations due to product mix, customer type, staffing differences, and exception handling.

**Preserve variants before summarizing:**

- Record the branch taken by each work unit and its trigger (for example, order type, customer tier, or product category).
- Use a Variant field or factual Notes in the event log; carry the distinction into the worksheet and map later.
- Record the key trigger for the variant (for example, "premium vs. standard service", "urgent vs. standard order").
- Time each variant path separately so you capture the actual cycle and wait times for each.

The number of loads is important context, but it is not automatically a different process variant. Orders WNF-001 through WNF-005 follow the same major path as one-load orders; they simply have two Wash records and two Dry records, one for L1 and one for L2. The Load ID prevents two machine cycles from being mistaken for one activity record while the Order ID preserves the fact that both belong to the same customer case.

At White Belt level, document variants explicitly during observation, but establish one primary baseline map for the first cycle. This keeps scope manageable while preserving evidence about meaningful differences across runs.

### 7.8. Record Who Owns Each Step

Record who performs each step so handoffs and role boundaries are explicit in your notes. If the 'who' changes from run to run, that is important data too. Best practice is to identify individuals by role or job title rather than by name. In very small businesses, you can get away with names, but as a business grows more complex this quickly becomes unworkable.

For Wash n' Fold, the customer and a worker participate in drop-off; cell staff perform the remaining activities. The Order ID remains attached through every handoff, and Load IDs identify the physical units handled by washers and dryers.

| Name | Type | Who |
|---|---|---|
| Drop-off and ticket | Step | Customer, Staff |
| Sort and tag | Step | Staff |
| Wash | Step | Staff |
| Dry | Step | Staff |
| Fold and package | Step | Staff |
| Stage and notify | Step | Staff |

### 7.9. Record Timestamps, Queues, Rework, and Interruptions

For each activity record, enter the Order ID, any applicable Load ID, activity, pass number, role/resource, Queue Entry, Setup Start, Work Start, and Work End. Record timestamps as events happen, including repeated activities. Keep participant explanations and hypotheses distinguishable from things directly observed. At this step, we tag three additional flow signals that often dominate lead time in practice: queues, rework loops, and interruptions. Your goal here is disciplined capture, not perfect diagnosis. Chapter 8 calculates and groups these records before Chapter 9 interprets the resulting baseline.

#### 7.9.1. Queue: Pure Waiting

**Definition 7.4.**

> **Definition: Queue**
>
> A queue is a state where work items sit idle, waiting for the next process step to begin. A queue adds wait time rather than processing time. Record all queue time in the event log's W/T calculation, record visible queue counts with their observation times, and draw a queue triangle when the waiting state is large or diagnostically important.

#### 7.9.2. Rework Loop: Work That Circles Back

**Definition 7.5.**

> **Definition: Rework Loop**
>
> A rework loop is a path where work returns to an earlier step because a defect, omission, or mismatch was found. Rework loops increase total effort and delay; they are a cost multiplier on total processing time and lead time and indicate quality problems. Add a row for each pass through the correction activity, linked to its work unit. On the worksheet, record the trigger, destination, and frequency with its denominator, such as 5 affected orders out of 39.

#### 7.9.3. Interruption: When Flow Stops

**Definition 7.6.**

> **Definition: Interruption**
>
> An interruption is a stop or diversion that pauses the normal flow of a work unit, usually to handle an urgent exception or priority task. Interruptions increase variation and make processing-time and completion-pace estimates less stable. Record the source, duration, and point in the activity in Notes so repeated patterns can be investigated later.

#### 7.9.4. Move from Observation to Event-Log Entry

The preceding definitions tell you what to notice. Now enter each observed activity record in the event log, preserving its parent Order ID, any applicable Load ID, activity name, pass number, role, timestamps, and factual context. Use Notes for the evidence behind a queue, rework, or interruption label; keep a participant's explanation or your own hypothesis visibly separate from direct observation.

At the end of each run, ask:

1. **Did any work have to cycle back?** If yes, record it in the Notes column with trigger and destination.
2. **Did anyone interrupt this step?** If yes, record the source and duration.
3. **Did work pile up before this step?** If yes, count the waiting units and record the observation time in Notes; label an estimate as an estimate.

After at least five runs:

- Tally how many times rework occurred and where it went back to.
- Note any recurring interruptions and their frequency.
- Identify the steps with the largest waits and visible queues.

Simple counts and percentages are useful at White Belt: distinguish five recorded corrections from five distinct affected orders. More detailed interruption-cost and variation analysis can follow at Yellow Belt.

For your own capstone observation, use the blank event-log template after you have chosen the boundary, unit of analysis, and timestamp convention. It supports absolute or relative timestamps and an optional child item identifier when work splits. Open the template in a browser, download the A4 color PDF, A4 B&W PDF, Letter color PDF, Letter B&W PDF, the CSV, or the editable workbook. Repeat the blank print page as needed. Keep the event log as your observation record; Chapter 8 supplies the separate sheets for calculations and summaries. Carry your boundary, clock convention, parent and child IDs, timestamps, and factual notes forward with the records you calculate.

> Practical Sampling Rule (White Belt)
>
> Collect timestamped observations from at least **5 runs** before summarizing a step's timings on the worksheet.
>
> - Five runs are a practice starting point, not proof that the process is stable or that the baseline is representative.
> - If timings or paths differ, keep observing and record the circumstances; no fixed small count guarantees that rare events or busy periods have been captured.
> - Keep context consistent: same process boundary, similar time windows, and similar staffing conditions.

At White Belt level, rough but consistently collected timings are sufficient. Precision matters less than a stable method.

We now have trustworthy timestamped records of what happened to individual Wash n' Fold orders. Chapter 8 begins by calculating the timing intervals those records support, then asks how to turn dozens of calculated rows into a process we can inspect as a whole.

### 7.10. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Understand) For each statement below, label it **Observed Condition** or **Assumed Cause**:

- "Orders wait 20 to 40 minutes before final packaging starts."
- "The staff is disorganized during shift handoff."
- "Three orders returned to the labeling step in one afternoon."

---

**2.** (Apply) Choose a process you know and draft a mapping boundary in two sentences:

1. sentence one defines start and stop events from the customer point of view,
2. sentence two defines the unit of analysis.

---

**3.** (Analyze) A team produced this observation snippet:

- Intake: 4 min work, 0 min wait
- Verification: 6 min work, 18 min wait
- Approval: 3 min work, 22 min wait

Which two delays should be prioritized first, and what does this suggest about flow quality?

---

**4.** (Analyze) A team recorded processing times for the same process step over five runs (in minutes):

$$2,\ 3,\ 4,\ 9,\ 2$$

1. Which record most needs additional context in the event log?
2. What should the observer check or record before deciding how to interpret it?
3. Should the team treat these five records as a stable baseline yet?

---

**5.** Apply Select one recurring process from work or daily life and complete the following before drawing anything:

1. Write the start event, stop event, and unit of analysis.
2. Prepare event-log fields for the case ID, any child item ID needed when work splits, repeated passes, roles, timestamps, and factual notes.
3. Identify one likely queue point and one likely handoff risk to verify during observation.

> **Note**
>
> In the next chapter, we group comparable event-log records onto the map worksheet and convert that evidence into a current-state map.

---

**6.** Apply Draft one observation note that is high quality and one that is low quality. Briefly explain why another reader can check the high-quality note.

---

**7.** Analyze An event log shows a long wait before approval but does not identify the roles on either side of the handoff. Explain what information is missing and why that weakens countermeasure design.

---

**8.** Understand Why should process variants be documented during observation even if only one primary baseline map is built first?

---

**9.** Analyze A team has five observations with one extreme outlier. Describe one decision rule for when to keep collecting versus when to use the observations as a provisional baseline.

---

### 7.10.1. Reflection Question

Write three or four sentences on one process where you might be tempted to infer causes too early, and how you will force yourself back to observed conditions first.

Think of a time when a record or direct observation contradicted someone's confident account of how work usually happened. How could you preserve both sources without treating either one as automatically correct, and what would you observe next?

**10.** [Stretch] (Evaluate) You must choose between two first-cycle observation approaches:

1. observe one process variant deeply,
2. observe three variants lightly.

Which approach is better for a first White Belt cycle, and why?

## 7.11. Chapter Summary

- Current-state mapping is a disciplined observation workflow, not a memory exercise or a polished drawing exercise.
- The Wash n' Fold boundary follows one customer order from drop-off through staging and readiness notification on an operating-time clock.
- Preserve individual activity records in the event log before calculating or summarizing the process on the worksheets.
- Queues, rework loops, and interruptions are not side notes; they are critical flow signals that often explain lead-time inflation better than the nominal step list alone.
- Chapter 8 calculates timing intervals from these individual records, then turns the results into the map worksheet and verified current-state process map.

## 7.12. Further Reading

Samuel H. Scudder's account of studying a fish under renowned naturalist Louis Agassiz provides a brief but brilliant account of the power—and challenge—of observing something scientifically.

# 8.

# Building the Current-State Process Map

> The map is not the territory. — Alfred Korzybski

## 8.1. Why This Chapter Matters

"Don't they make a tool for this?" is a common reaction when process mapping gets messy. Tools help, but the real bottleneck is usually the quality of your observations, not software. The completed event log introduced in Chapter 7 preserves what happened to individual Wash n' Fold orders and to the loads created from them. That evidence is trustworthy, but dozens of rows still do not give us a picture we can inspect all at once. This chapter shows how to summarize those records without losing their provenance, turn the summaries into a current-state map, and verify that the model represents the work fairly.

## 8.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Lean Six Sigma Principles and Tools Classify each process step as value-added (VA), required non-value-added (RNVA), or non-value-added (NVA) from the customer's point of view **Analyze**
- Data Understanding Aggregate comparable records using count, sum, and mean while preserving the observation unit, and explain what may be concealed when a mean is transferred to a process map **Analyze**

- Data Understanding Convert a completed map worksheet into a current-state SPPM that represents supported queues, release delays, handoffs, and rework without presenting hypotheses as observations (**Apply**)
- Data Understanding Verify a first-pass map with a process participant and record accepted or rejected revisions with reasons (**Evaluate**)

Chapter 7 laid out the complete nine-step workflow and produced timestamped Wash n' Fold observation records through Step 5. We begin with Step 6: calculate the timing intervals those records support. Then we carry the calculated evidence through the worksheet, visual map, and participant verification.

## 8.3. Wash n' Fold Project: From Records to a Map

The event log follows 39 customer orders through the approved drop-off-to-notification boundary. It contains one order-level record for activities that act on the whole order and one record per physical load for Wash and Dry: 65 Wash records and 65 Dry records in all. Parent Order IDs, child Load IDs, roles, timestamps, and five recorded Fold corrections preserve the connections among those records. We will use those records to decide what can be combined, what must remain visible, and what the resulting map does—and does not—allow us to conclude.

## 8.4. Calculate Each Activity Record's P/T, W/T, and S/T

The timestamp columns in the event log contain observations. Processing time, wait time, and setup time are derived values calculated from those observations on the timing calculation worksheet:[1]

$$\text{W/T} = \text{Setup Start} - \text{Queue Entry}$$

[1]The terms in this book follow the value-stream mapping convention used by Rother and Shook: **processing time (P/T)** is the time a unit is actually being processed, **cycle time (C/T)** describes the pace of successive completions from a process or operator, **changeover time (C/O)** is the time required to prepare for a different product, service, or operating mode, and **lead time (L/T)** is elapsed time across a stated boundary (Rother and Shook 1999). This book uses **setup time (S/T)** for the broader preparation interval before processing begins and reserves C/O for a genuine change from one product, service, or mode to another. The terms are overloaded in the operations literature. Hopp and Spearman use **cycle time** for the elapsed time a job spends in the system and **lead time** for a planning allowance (Hopp and Spearman 1996). When reading another source, compare its definitions and time boundaries before comparing its numbers with ours.

$$\text{S/T} = \text{Work Start} - \text{Setup Start}$$

$$\text{P/T} = \text{Work End} - \text{Work Start}$$

A missing timestamp is unknown, not zero. When an activity has no setup, Setup Start equals Work Start and its S/T is zero.

### 8.4.1. Know the Three Timing Labels

The log captures four timestamps: **Queue Entry**, **Setup Start**, **Work Start**, and **Work End**. They mark three distinct intervals. Learn the labels before calculating them on the worksheet or using them in the map:

**Definition 8.1.**

> **Definition: Processing Time**
>
> **P/T** means processing time. For one pass through an activity, processing time is the elapsed time from Work Start to Work End while the unit is actively being processed. **Average processing time** is the arithmetic mean across comparable activity records. Write Avg P/T when transferring that mean to the worksheet or map.

**Definition 8.2.**

> **Definition: Wait Time**
>
> **W/T** means wait time. For one pass through an activity, it is the elapsed time from queue entry until setup starts; if there is no setup, it ends when work starts. **Average wait time** is the arithmetic mean of those observed waits; label the summary Avg W/T. All queue time is wait time, whether or not you later choose to draw that queue as its own symbol on the map.

**Definition 8.3.**

> **Definition: Setup Time**
>
> **S/T** means setup time. For one pass through an activity, setup time is the preparation interval from Setup Start to Work Start. **Average setup time** summarizes comparable observed preparation intervals; label it Avg S/T. Record it separately from queue wait time. Record `0` only when no setup occurs; leave unobserved setup time unknown.

**Definition 8.4.**

> **Definition: Changeover Time**
>
> Changeover time, written **C/O**, is the setup time required to switch a process from one product, service, batch type, or operating mode to another. Every changeover is setup, but not every setup is a changeover. Use C/O only when the observation records an actual transition and identifies what changed.

For one load of laundry waiting to be washed, the three labels describe different parts of its visit to the washing step:

- **W/T:** The load enters the washing queue at 10:00, but the worker does not begin preparing the washer until 10:15. Its wait time is 15 minutes.
- **S/T:** The worker spends 3 minutes gathering supplies, checking the machine, and selecting the load's settings. Its setup time is 3 minutes. If the record separately established that the worker was switching the washer from a different mode, that transition could be reported as C/O.
- **P/T:** Once the washer starts, the wash cycle runs for 25 minutes. Its processing time is 25 minutes, even though the worker can attend to other tasks while the machine runs.

For these recorded intervals, waiting ends when setup starts, setup ends when work starts, and processing time ends when the work for that pass finishes. Automatic machine processing can count as P/T even when no worker is occupied throughout.

**Figure 8.1.:** The four timestamps in one activity record divide a pass through that activity into wait, setup, and processing-time intervals.

The completed Wash n' Fold record makes the distinction visible: its timestamps are the evidence, while `W/T = 90.8`, `S/T = 1.6`, and `P/T = 29.4` minutes are calculations based on that evidence. Keep each calculated value connected to the same record before grouping comparable rows.

Use the **Mapping Worksheet Set** to make that connection visible on paper before asking a spreadsheet, a script, or a process-mining tool to do it at scale. It contains two physical worksheets with one job each: calculate timing values for individual records, then summarize comparable calculated records for the map.

- Download the editable Mapping Worksheet Set (XLSX)
- Download the Timing Calculation Sheet as an A4 color PDF, A4 B&W PDF, Letter color PDF, Letter B&W PDF, or CSV.
- Download the Map Summary Sheet as an A4 color PDF, A4 B&W PDF, Letter color PDF, or Letter B&W PDF.
- Download the Map Summary Sheet CSV.
- Inspect the short worked example as an editable workbook or printable worksheet set.
- Keep the full completed references available: Mapping Worksheet Set XLSX, Timing Calculation Sheet in a browser or PDF, and Map Summary Sheet in a browser or PDF.

Start with the five records in the short event-log excerpt: activity `Wash`, Child ID `L1`, and Parent IDs `WNF-001` through `WNF-005`. Transfer their Parent ID, Child ID, timestamps, rework flag, and factual notes from the event log to the Timing Calculation Sheet. Calculate W/T, S/T, and P/T for each row, keeping every result beside the timestamps that support it. Then transfer the five calculated rows to one Map Summary Sheet row as `n`, timing sums, and averages, with a note about what varies or needs verification. Compare that result with the short worked copy before opening the full references.

The timing workbook expects numeric elapsed minutes from one stated origin. If your event log uses absolute Excel date/times, first convert each timestamp to elapsed minutes with `(timestamp - clock origin) * 1440`. That converts Excel's days to minutes while retaining the original clock in the log. Each duration formula checks only its own two endpoints: knowing Work Start and Work End lets you calculate P/T even if Queue Entry is unknown. For more records, add rows within the workbook's timing table and copy the W/T, S/T, and P/T formulas into those rows. The printable sheets leave room for writing; repeat a blank page when you need more records.

Next, repeat the timing calculation for the complete parent order `WNF-003`, using its source excerpt (CSV) or PDF. Its order-level records, compatible

load records, and Fold correction make a useful check on the distinction between a record-level calculation and an order-level story. Compare your nine rows with the completed timing sheet (PDF) or CSV. Do not sum its child-load timings and call the result its elapsed lead time. Chapter 9 returns to the parent-order projection when that is the question being asked.

### 8.5. Build the Map Summary Sheet

Step 7 moves from calculated individual activity records to activity summaries. Group records with the same activity and a comparable process path and operating context. They must also describe the same kind of observed unit. Do not average order records and load records together or hide a meaningful variant in one row.

The Map Summary Sheet is the second worksheet in the set. For each map candidate, it records the activity or queue, observation unit, *n*, timing sums and averages, role, value classification, type, variation or evidence note, and a clear instruction for the eventual map. The completed Wash n' Fold reference shows the target result without replacing the arithmetic you perform on the five-record practice group. The compact Process Map Worksheet is an alternative paper summary after the event log and timing calculations are complete. It summarizes activities; it does not replace the observation record.

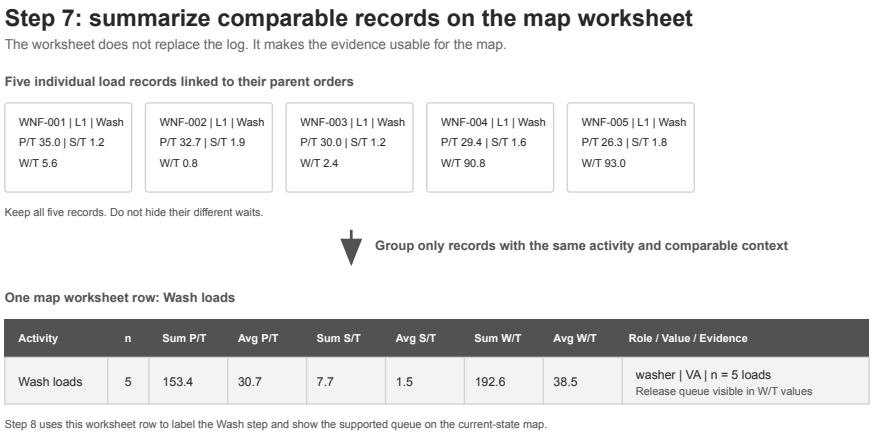



**Figure 8.2.:** Step 7 turns five comparable calculated load records into one Map Summary Sheet row without losing their parent Order IDs.

Each named activity receives its own row. Record roles rather than employee names so ownership remains understandable when staffing changes.

**Definition 8.5.**

> **Definition: Decision**
>
> A decision identifies the point where a process can follow alternative paths. If making the decision consumes work or waiting time, preserve that time in the log.

The count $n$ records how many comparable activity records support a timing summary. State what those records represent: $n = 39$ orders is not interchangeable with $n = 65$ loads. For each timing column, count the observations, add them while preserving the recorded time unit, and divide the sum by $n$. For example, if five comparable waits sum to 27 minutes, then $n = 5$ and Avg W/T $= 27/5 = 5.4$ minutes. Record the sum as well as the mean so another reader can check the arithmetic. Round only the final mean, not the individual observations or partial sum; unless the project requires another convention, report one more decimal place than the original whole-unit observations and otherwise retain the precision of the recorded timestamps. These are summaries of the recorded observations, not additional stopwatch readings.

If different timing columns have different numbers of complete observations, state their counts separately. Notes record the source log/window, units, classification reasons, variants, rework trigger and destination, and questions for participant verification. A repeated activity can produce more records than there are distinct work units.

After the timing evidence, record Value as VA, RNVA, or NVA from the customer's perspective and Type as `step`, `queue`, or `decision`. A step is work, a queue is a waiting state, and a decision identifies alternative paths. These classifications still matter, but they do not replace the counts, sums, and averages that support the map.

### 8.5.1. Classify Value Type (VA/RNVA/NVA)

Before classifying activities, agree on the value-classification rules. Without a shared convention, two people can map the same process and produce incompatible classifications.

Use the customer's point of view as the filter and apply the same three-part test to every step.

**Definition 8.6.**

**Definition: Value-Added Process Step**

A process step is a value-added process step only when it passes this **three-part test**:

- **(i) Customer value:** The step directly creates or advances an outcome the customer receives and values.
- **(ii) Transformation:** The step changes the form, fit, function, information, or condition of the product or service toward that outcome.
- **(iii) Right first time:** The step is performed correctly without creating a need for correction or rework.

**Definition 8.7.**

**Definition: Non-Value-Added Process Step**

A non-value-added process step fails one or more parts of the value-added test and is not currently required to deliver the product or service safely and reliably.

**Definition 8.8.**

**Definition: Required Non-Value-Added Process Step**

A required non-value-added process step fails one or more parts of the value-added test but is currently required by law, regulation, contract, a documented safety or control need, or another necessary condition of delivering the service reliably. Record the specific requirement rather than treating RNVA as a permanent exemption from improvement.

A Word on the Human Side

Value classification can feel personal to operators and managers. When facilitating this step, emphasize that “non-value-added” describes the activity from the customer’s perspective, not the worth of the person doing the work.
It is very important not to make people feel like you are diminishing the importance of their contributions.

Every process step must fall into exactly one of these three exhaustive and mutually exclusive categories:

- **VA (value-added):** steps that pass all three parts of the customer-value, transformation, and right-first-time test. Examples: machining a part, cleaning garments, completing a clinical procedure.
- **RNVA (required non-value-added):** steps required by regulation, contract, billing, or safety controls. Examples: identity verification, compliance documentation, payment capture.
- **NVA (non-value-added):** steps that add no customer value and are not required. Examples: waiting in queue, re-entering duplicate data, moving work because of poor layout.

> Pro Tip: Document Why a Step is Required
>
> In the Notes column, record exactly *why* a RNVA step is required. If a specific policy or regulation requires it, find the reference. This matters for three reasons.
>
> - **Documentation quality.** Understanding the regulatory context makes it much easier to periodically review for policy changes.
> - **Policies change.** A step that was once required may no longer be—or new requirements may have been added that you aren't currently meeting. Compliance is a moving target; always do your homework.
> - **Some "policies" are not actually policies.** Certain bosses speak with such commanding air that their personal preferences solidify in the minds of employees—and themselves—as official policy, when in fact they are nothing of the sort. If you stick around process improvement long enough, you'll eventually get to have a really fun one-on-one with somebody's boss's boss about one of these "shadow" policies.

### 8.5.2. Classify the Wash n' Fold Activities

The customer receives clean, dry, folded laundry. Wash, Dry, and Fold and package directly transform the order and are VA. Drop-off and ticket, Sort and tag, and Stage and notify are required for identification, safe processing, and return, so this case treats them as RNVA. Waiting and correction work are NVA.

| Activity | Role | Value | Type | Case-specific reason |
|---|---|---|---|---|
| Drop-off and ticket | Customer, Staff | RNVA | Step | Establishes order identity, requirements, and traceability. |
| Sort and tag | Staff | RNVA | Step | Protects the order and determines compatible washing groups. |
| Wash | Staff | VA | Step | Cleans the laundry. |
| Dry | Staff | VA | Step | Dries the laundry. |
| Fold and package | Staff | VA | Step | Produces the folded, packaged result. |
| Stage and notify | Staff | RNVA | Step | Makes the finished order findable and communi-cates readiness. |

### 8.5.3. Aggregate Comparable Activity Records

First choose the five comparable Wash rows you calculated on the Timing Calculation Sheet. Using those P/T, W/T, and S/T values, enter the count, sum, and average for each timing label on the Map Summary Sheet. For P/T, the count is 5, the sum is 153.4 minutes, and the mean is $153.4/5 = 30.68$, displayed as 30.7 minutes. Write a variation or evidence note before drawing anything: five rows are practice evidence, not proof that all Wash loads behave alike. That five-row exercise does not replace the complete 65-load summary.

After completing that small exercise, use the supplied activity summary as the worked reference for all 39 orders and their 65 loads. The Map Summary Sheet preserves the count, sum, and mean for P/T, W/T, and S/T because those are the values needed to check and label the map. Its variation or evidence note should still record anything the mean may conceal, such as a wide spread, an unusual observation, or a changing queue. Chapter

9 returns to the underlying observations and teaches how distributions, ranges, medians, and percentiles expose those patterns more directly.

The process map follows customer orders, so its Batch Release queue uses a separate order-level projection. For each order, measure from Wash queue entry to the first load's setup start; across 39 orders, that mean is 501.7 minutes. Do not average 65 load waits and label the result as 39 customers' average wait.

The completed main-path timing summary is below; all sums and averages are in minutes. The role, value, type, and classification reasons remain in the context table above. The observation basis is explicit because Wash and Dry contain 65 load records while the other main activities contain 39 order records.

**Table 8.1.:** Main-path activity summary from Friday 91.

| Activity | n | Sum P/T | Avg P/T | Sum S/T | Avg S/T | Sum W/T | Avg W/T | Rework trigger? |
|---|---|---|---|---|---|---|---|---|
| Drop-off and ticket | 39 | 115.2 | 3.0 | 0.0 | 0.0 | 197.8 | 5.1 | N |
| Sort and tag | 39 | 187.6 | 4.8 | 0.0 | 0.0 | 174.1 | 4.5 | N |
| Wash loads | 65 | 1,977.2 | 30.4 | 96.9 | 1.5 | 32,560.1 | 500.9 | N |
| Dry loads | 65 | 2,944.9 | 45.3 | 42.5 | 0.7 | 51.8 | 0.8 | N |
| Re-unite, fold, and package | 39 | 630.8 | 16.2 | 19.5 | 0.5 | 139.4 | 3.6 | Y |
| Stage and notify | 39 | 155.0 | 4.0 | 0.0 | 0.0 | 1,479.1 | 37.9 | N |

The five Fold correction records form an observed NVA rework path. Their mean P/T is 6.8 minutes and mean W/T is 5.6 minutes.

### 8.6. Draft the Visual Map (SPPM Conventions)

When translating the map worksheet into a diagram, use one symbol system consistently. SPPM is our teaching label for a simplified set of widely used workflow and value-stream conventions. It is not a brand-new notation invented from scratch.

| Symbol | Shape / Line | Meaning |
|---|---|---|
| Process step | Rectangle | A unit of work performed by a role; colored by value type (Green = VA, Gray = RNVA, Red = NVA) |
| Data box | Small box below step rectangle | Records role, Avg P/T/S/T/W/T, and sample count or a shared evidence reference |
| Queue / waiting state | Triangle (light orange) | A queue of work items waiting for the next step; size or label states a time-stamped queue count or average wait time |
| Subprocess | Double-line rectangle | A step that can be exploded into its own detailed map; see separate detail map |
| Decision / approval | Diamond (light yellow) | A branch point (approval, check, gate) where variants diverge |
| Start / stop event | Rounded rectangle | Scope boundary from the customer's perspective |
| Forward progress | Solid arrow | Normal flow from one step to the next |
| Rework loop | Dashed arrow | Work returns to an earlier step due to defect or mismatch; label with rework rate (e.g., "5%") |

When a wait moves from a worksheet row into its own queue symbol, count it only once. A map caption should state units, the observation window, and that displayed timings are means. Keep the detailed records and worksheet beside the map; readers need a legible view of the flow without losing access to its evidence.

Use the customer-facing boundary unless you explicitly declare a narrower scope. We will now apply this visual grammar to Wash n' Fold, including data boxes, queue triangles, and rework notation.

### 8.7. Select Queues and Draw the Current-State Map

Step 8 chooses which worksheet evidence must remain visible on the map.

Every W/T remains in the evidence table, but not every small delay needs its own map symbol. The 501.7-minute order-level mean Batch Release queue before Wash dominates the current-state picture. The 37.9-minute Notification Batch queue is also persistent and meaningful. We promote both to explicit queue triangles.

Inspect the order-level Batch Release projection (CSV) or its printable derivation. It keeps one row per order: the earliest Wash setup start minus the earliest Wash queue entry for that order. The mean of those 39 waits is 501.7 minutes; the mean across 65 load waits is 500.9 minutes. The map uses the order-level queue label because its boundary follows a customer's order. It does not add that label to the load-level wait as though they were two separate delays.

Five of 39 orders failed the packaging check and entered Fold correction, so the map includes an evidenced correction branch. No recurring machine breakdown or other interruption appears in the selected observation window; those remain **Not evidenced** rather than being added from speculation.

#### 8.7.1. Draw the Selected Evidence

We now translate the verified evidence into a Simplest Possible Process Map (SPPM). The task boxes carry mean P/T and S/T with their observation basis. The two major queue triangles carry mean W/T; the correction branch has its own wait and n = 5. Here *n* counts activity records. On this Friday, each affected order has one recorded correction. The correction branch records its observed frequency, while the smaller waits remain available in the completed table.

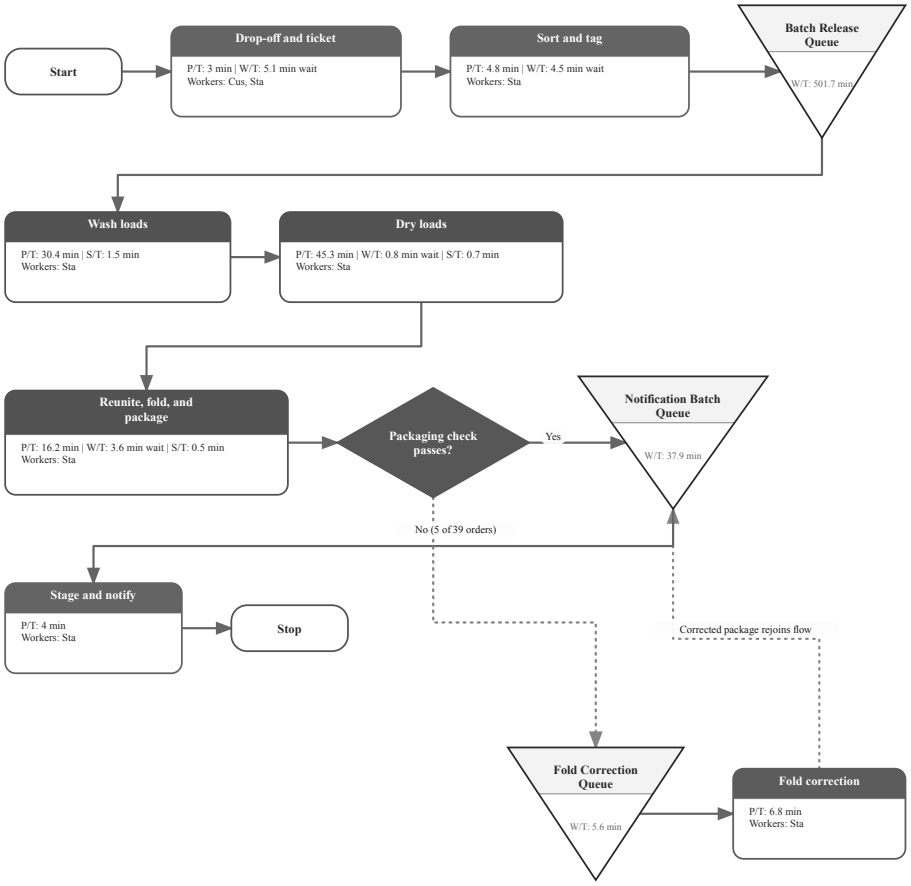



**Figure 8.3.:** Wash n' Fold current-state process map (SPPM)

Download the printable current-state map to keep beside the worksheets.

All P/T, S/T, and W/T labels in this map mean averages from Friday 91. Wash and Dry summarize 65 load records; the other main activities and the Batch Release queue summarize 39 orders; correction work has $n = 5$. A promoted wait belongs to its queue symbol, not a second copy inside its task box. Minor waits remain in the worksheet and must be included in process-level calculations.

The map is intentionally an averages-level summary. It does not replace the event log or claim that every order waited exactly the same amount of time.

### 8.8. Verify the Map with a Participant

Before baseline analysis, review the draft map with at least one process participant and, when those are different people, the process owner. Ask the reviewer to walk one recent work unit through the drawing rather than merely asking whether the map "looks right." Record each proposed correction and whether you accepted or rejected it, with a brief reason. Use the blank **Participant Verification Record** as a PDF, XLSX, or CSV. It records the questioned feature, observed evidence or participant explanation, proposed revision, decision, reason, and the log, worksheet, or drawing updated.

Use this checklist during the review:

- **Boundary drift:** start/stop points change mid-analysis.
- **Averaging too early:** high variation gets hidden in one number.
- **No wait-time capture:** map shows activity but not delay.
- **Policy mapping:** documented process, not observed process.
- **Missing ownership:** handoffs are not explicit.
- **No rework notation:** loops vanish from the visual.
- **Variant representation:** decision points and branch paths are marked where variants occur.

Step 9 tests the first-pass representation against participant knowledge and the supplied records.

To see why participant verification matters, suppose the team first sketches Wash and Dry as though each order moved through them as one machine load. That drawing conflicts with the event log, which contains separate L1 and L2 records for many orders. During verification, the operator explains that lights, darks, fabric-care needs, soil level, and order weight may split an order into compatible loads. Those loads are never mixed with another customer's order and must reunite before folding. The data support the correction: 26 of the 39 observed orders used two loads, producing 65 Wash and 65 Dry records.

We keep a revision record instead of silently redrawing the map:

| Proposed revision | Decision | Reason |
|---|---|---|
| Show compatible loads through Wash and Dry, then reunite before folding. | Accepted | Operator explanation agrees with the order context in the event log. |
| Add a Fold correction branch. | Accepted | Five orders contain explicit correction evidence. |

| Proposed revision | Decision | Reason |
|---|---|---|
| Promote the Batch Release and Notification Batch waits to queue symbols. | Accepted | Both delays are persistent and material in the timing summary. |
| Extend the boundary through customer pickup and payment. | Rejected | Pickup, payment, and customer travel are outside the approved improvement boundary. |
| Add a machine-breakdown interruption. | Rejected | No recurring breakdown is evidenced in this observation window. |

The worked **Participant Verification Record** is available as a PDF, XLSX, or CSV. It expands this case review with evidence and revision references; it is not a record of an actual interview.

With participant verification complete, the map is an evidence-backed current-state baseline rather than a polished guess. Chapter 9 will use the order-level records to study distributions, P90 lead time, WIP, throughput, and the 5:00 p.m. customer promise. Carry the verified map, revision record, worksheets, and original log into that analysis.

> **!** Mini-Workshop: Map the Current Process
>
> Using the completed table and the SPPM conventions, sketch the current process on paper or in a drawing tool. Include:
>
> - the drop-off-to-ready-notification boundary,
> - one customer order as the unit of analysis,
> - the six main activities,
> - compatible loads that reunite before folding,
> - the Batch Release and Notification Batch queues,
> - the evidenced Fold correction path,
> - role, mean P/T, mean W/T, and mean S/T where appropriate.
>
> Then answer:
>
> 1. Which work is clearly value-added?
> 2. Which queue dominates the current state?
> 3. What main-path lead-time estimate do the displayed means imply?
> 4. What information does that estimate conceal?
> 5. Which delay should the team investigate first, without yet claiming a root cause?

Sample Answers

1. Wash, Dry, and Fold and package directly transform the laundry and are VA.
2. The Batch Release queue before Wash dominates, with a mean W/T of 501.7 minutes.
3. The map that combines load-level and order-level averages does not support a valid lead-time estimate by simple addition. Use the order summary instead: mean P/T is 104.5 minutes, mean W/T is 554.0 minutes, and mean S/T is 4.1 minutes, for a mean operating lead time of 662.6 minutes including observed rework.
4. The estimate conceals large order-to-order variation, the queue's growth over the day, different weights and load counts, and the five correction cases.
5. Investigate the serial batch-release rule and the queue it creates before Wash.

## 8.9. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** Analyze Using the completed Wash n' Fold table, explain why Wash, Dry, and Fold and package are VA while Drop-off and ticket, Sort and tag, and Stage and notify are RNVA. Identify one NVA queue and one NVA rework activity from the same evidence.

---

**2.** Analyze A small dental practice handles patient appointments as follows: the room is cleaned and prepared, the patient checks in at reception, waits in the waiting room, is called to the exam room, the hygienist performs the cleaning, the dentist reviews the x-rays, performs the exam, and the patient checks out and schedules the next appointment.

Using SPPM conventions, classify each step as VA, RNVA, or NVA and briefly justify each classification.

---

**3.** Apply Write a one-paragraph evidence summary of the representative Wash n' Fold Friday supplied by the simulation. Include the boundary, unit of analysis, largest queue, one rework observation, and one investigation target.

---

**4.** Evaluate A teammate proposes skipping direct observation and building the map from interviews to save time. Decide whether this is acceptable for a White Belt baseline, justify your decision, and give two evidence-quality risks.

---

**5.** Apply Point to where each artifact appears in the Wash n' Fold application and explain what it contributes:

(a) process boundary statement,
(b) learner event log,
(c) completed activity summary,
(d) verified current-state map.

Then explain why the chapter builds them in that order.

---

**6.** Apply A small print shop handles rush poster orders with this observed flow:

1. Receive order details and file from customer (4 min)
2. Queue before design check (18 min)
3. Design check and print setup (10 min)
4. Queue before printer availability (25 min)
5. Print run (12 min)
6. Trim and package (6 min)
7. Queue before pickup notification (20 min)
8. Send pickup notification (3 min)

Apply the SPPM mindset:

(a) Name the process boundary.
(b) Identify the two biggest queues.
(c) Calculate a first-pass total lead-time estimate.
(d) Suggest one practical target for the first PDCA cycle.

---

**7.** (Apply) For the print-shop flow in Q6, classify each queue as likely release-control delay or likely capacity-constraint delay. State one observation you would collect next to validate each classification.

---

**8.** (Analyze) A team map shows no rework path, but three orders were relabeled in one shift. Explain why this is a mapping defect and identify how the rework should be represented and quantified.

---

**9.** (Evaluate) Review this draft: "Each customer order uses one washer and one dryer; there is no correction path; Stop means pickup and payment." The operator says, "Some orders need compatible loads that reunite before folding. Keep pickup outside our improvement boundary. I also think breakdowns must be a major problem." The log records 26 two-load orders and five corrections among 39 orders, with no recorded breakdowns. Produce a revision record with proposed change, accepted/rejected/not yet supported, supporting evidence, and one follow-up question.

---

**10.** (Analyze) Compare a Wash n' Fold first-pass draft that treats every order as one load with the verified map. Identify two differences and explain why they matter for improvement decisions.

---

**11.** (Apply) Record or construct five event-log rows for one activity in a process of your choice. Calculate P/T, W/T, and S/T from displayed timestamps, then calculate the count, sum, and mean for P/T. State which values belong on the Map Summary Sheet, which value belongs on the map, and what warning should accompany it.

---

### 8.9.1. Journal and Reflect

Describe a time when an average gave you a misleading impression of a process. What individual observation or participant disagreement would make you revise the map, and how would you preserve that evidence?

Uncertain arrivals, order weights, sorting needs, and load counts all remain present in this case. Why might that uncertainty harm a serial batch/push system more than a controlled FIFO pull system? Write 3–4 sentences without assuming that improvement requires eliminating the uncertainty itself.

### 8.10. Chapter Summary

- Step 7 groups comparable Wash n' Fold activity records onto the map worksheet while preserving their source, context, counts, sums, and averages.
- Value classification distinguishes customer-valued transformation from required support work, waiting, and correction without judging the people doing the work.
- The 501.7-minute order-level Batch Release queue is the dominant current-state delay, while the 37.9-minute Notification Batch queue is a smaller persistent delay.
- Participant verification corrected a one-order/one-load drawing: compatible loads remain tied to one order and reunite before folding.
- Five observed Fold corrections support a rework path; unsupported breakdowns remain Not evidenced.

# 9.

# Measuring and Interpreting Process Performance

> A summary can be accurate and still conceal the thing you most need to see.

## 9.1. Why This Chapter Matters

Suppose someone gives you four datasets and reports the following facts about each one:

- the mean of $x$ is 9.0,
- the mean of $y$ is 7.5, and
- the same straight line describes the average relationship between $x$ and $y$.

Would you expect the observations to look alike?

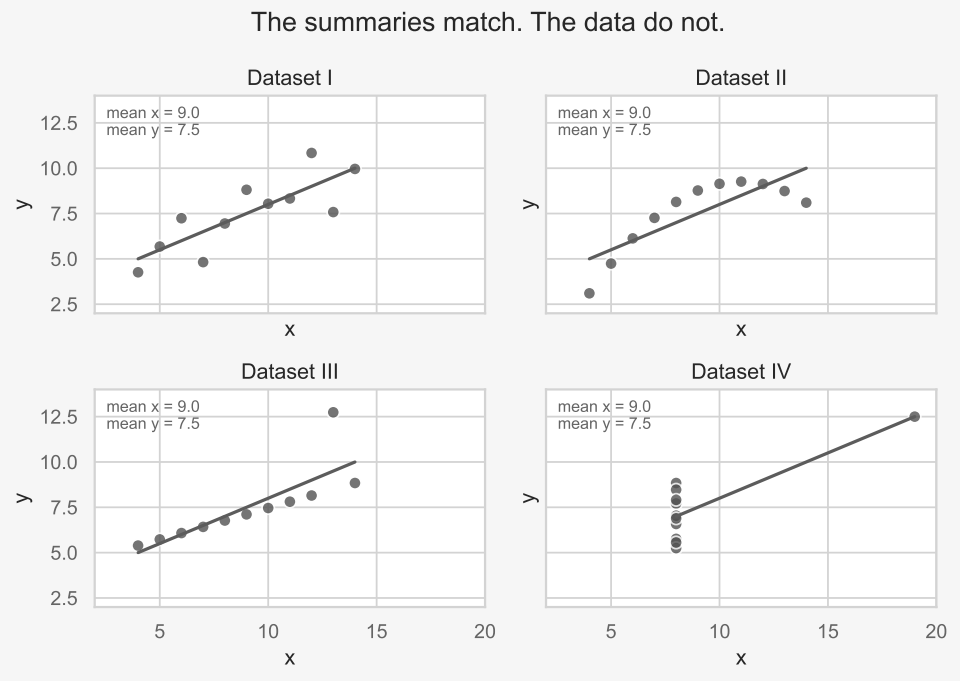



**Figure 9.1.:** Anscombe's quartet: nearly identical numerical summaries can conceal very different patterns.

They do not. One dataset follows the line reasonably well, one bends, one is disrupted by an unusual observation, and one depends almost entirely on a single distant point. These are Anscombe's quartet, constructed to demonstrate why numerical summaries should be inspected alongside the observations they summarize (Anscombe 1973).

The lesson is not that averages are bad. The lesson is that every summary answers a limited question. Before trusting one, we need to know what was observed, how the observations are distributed, and what decision the summary is supposed to support.

Wash n' Fold now presents the same problem in operational form. The event log contains hundreds of timestamps, the map displays a few averages, and customers experience individual orders rather than averages. This chapter turns those records into a baseline we can inspect and use.

## 9.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Data Understanding Explain how a distribution describes the observed values of a quantity and how often or where those values occur **Understand**

- Data Understanding Aggregate an event log into activity and order summaries using count, sum, minimum, maximum, range, mean, median, and mode while preserving units and the group being summarized **Apply**
- Statistical Analysis Calculate and interpret takt time, processing time, cycle time, operating lead time, and PCE using consistent boundaries and explicit treatment of rework **Apply**
- Statistical Analysis Interpret tabular and graphical evidence to compare typical, tail, and threshold performance using mean, median, nearest-rank percentiles, and the customer-promise violation rate **Analyze**
- Statistical Analysis Use WIP, throughput, and Little's Law with a matching boundary and observation period, distinguishing average WIP from a snapshot **Apply**
- Lean Six Sigma Principles and Tools Connect a Wash n' Fold metric symptom to a first-pass waste hypothesis and the evidence needed to test it **Analyze**

### 9.3. From Events to Customer Observations

Chapter 7 introduced the event log as the foundational record. Chapter 8 calculated each activity record's P/T, W/T, and S/T, grouped comparable records on the Mapping Worksheet Set, and transferred selected averages to the current-state map. That compression made the flow readable, but it also hid the individual observations.

Now we return to them. We need to know how the process behaved across complete customer orders, whether the average represents most customers reasonably well, and which evidence should direct the next investigation. All calculations in this chapter can be done on paper, with a basic calculator if useful. The charts help you inspect the evidence; you are not expected to create them at White Belt.

| Artifact | Question it answers | What one row represents |
|---|---|---|
| Event log | What happened? | One order- or load-level activity record linked to its parent case |
| Map worksheet | What does each activity look like across observations? | One activity with counts and timing summaries |
| Process map | How does work move and where does it wait? | Selected activities, queues, and paths |
| Order summary | What happened to this customer's order? | One complete order |
| Performance readout | How did a defined group of orders perform? | A metric with its group, units, and period |

These artifacts do not compete. The log preserves individual events, the worksheet supports the map, the order summary reconstructs each customer's experience, and the performance readout brings those measures together so we can decide what to investigate.

> **Evidence and Clock**
>
> We continue with current-state Friday 91 from Chapter 8: 39 orders, including 21 dropped off by 9:00 a.m. One customer order is the unit of analysis; its boundary runs from drop-off through staging and readiness notification. Pickup and payment remain outside it.
> Minute 0.0 is 7:00 a.m.; minute 120 is 9:00 a.m.; minute 600 is the 5:00 p.m. promise; minute 690 is the 6:30 p.m. close. The model follows unfinished orders on a cumulative operating-time clock until all are ready. Closed hours are omitted. These are operating lead times, not overnight calendar durations.

- Download the activity event log
- Download the aggregated activity worksheet data
- Download the learner order summary

Keep your Chapter 8 map and Participant Verification Record beside these tables. For a first complete-order check, use `WNF-003`: its nine activity records include two compatible Wash loads, two Dry loads, and one Fold correction. Read its timestamps in the source excerpt, compare them with the worked timing sheet, and then locate its single row in the learner order summary. Its drop-off-to-readiness operating lead time is 166.3 minutes, rather than the sum of every child-load duration.

The order summary derives its durations from the displayed one-decimal event-log timestamps. This keeps the hand calculations reproducible.

### 9.4. What Are We Actually Summarizing?

Aggregation means combining observations into a summary. Before calculating, complete this sentence:

> This number describes ___ for ***, measured in*** , across ___.

For example, "30.4 minutes" is incomplete. "Mean Wash P/T was 30.4 minutes per load across 65 observed loads" identifies the quantity, aggregation, physical unit, observation unit, and group. It is still not the duration of one particular load.

The same log supports several groupings: by load for machine activity, by activity for mapping, and by order for customer lead time. A sum of all event-log rows is not automatically the elapsed time of one order or the shop's working day.

### 9.4.1. A Distribution Is the Pattern of Observations

Take the five L1 Wash P/T observations selected for Chapter 8's short practice excerpt, `WNF-001` through `WNF-005`, in minutes:

$$35.0,\ 32.7,\ 30.0,\ 29.4,\ 26.3$$

These five values form an observed **distribution**. A distribution describes which values were observed and how often or where those values occur. With five values, we can inspect every observation directly. With 65 loads or hundreds of customer orders, a graph often makes the shape, gaps, clusters, and unusual values easier to see than a long table does.

Exploratory data analysis begins with that act of inspection. We look before deciding which summaries are adequate, which comparisons are fair, or which observations deserve another question.

### 9.4.2. Count, Sum, and Mean

**Count** tells us how many observations there are: $n = 5$. **Sum** adds their values: $35.0 + 32.7 + 30.0 + 29.4 + 26.3 = 153.4$ minutes. **Mean**, or arithmetic average, divides the sum by the count:

$$\begin{aligned} \text{Mean} &= \frac{\text{Sum}}{\text{Count}} \\ &= \frac{153.4}{5} \\ &= 30.68 \approx 30.7 \text{ minutes} \end{aligned}$$

The sum is the total of five load-level processing durations. It is not 153.4 minutes on one washer's clock: the loads belong to different orders and may occupy different machines in parallel. For the full 65-load Wash group, the P/T sum is 1,977.2 minutes, giving a mean of $1,977.2/65 \approx 30.4$ minutes.

Use the mean when the total burden is meaningfully distributed across the observations: average work per load, average lead time per order, or average demand per day. Do not ask it to show the shape of the distribution or the experience of every customer.

### 9.4.3. Minimum, Maximum, and Range

Sort those five observations:

$$26.3,\ 29.4,\ 30.0,\ 32.7,\ 35.0$$

The **minimum** is 26.3 and the **maximum** is 35.0. They describe the shortest and longest observed durations; they are not guaranteed operating limits. Their difference, $35.0 - 26.3 = 8.7$ minutes, is the observed **range**.

The range answers a practical first question: how far apart were the shortest and longest observations? Because it depends only on those two observations, one extreme value can change it sharply.

### 9.4.4. Median and Percentiles

The **median** is the middle sorted observation: 30.0 minutes. With an even count, average the two middle observations. For example, the median of $26.3, 29.4, 30.0, 32.7$ is $(29.4 + 30.0)/2 = 29.7$.

The mean uses every value, so one very long record can pull it upward. The median describes the middle position. Neither describes every customer's experience.

A **percentile** is also a position in a sorted distribution. The 25th percentile, or P25, marks the lower quarter; P50 is the median; P75 marks the upper quarter; and P90 marks the slow tail used later in this chapter.

The conventional **five-number summary** is:

1. minimum,
2. first quartile, $Q_1$ or P25,
3. median, or P50,
4. third quartile, $Q_3$ or P75, and
5. maximum.

It gives a compact account of position and spread. It does not include the mean, count, or sum, and it does not replace inspection of the observations. Software uses several percentile conventions, so record the convention when the choice can change the reported value.

### 9.4.5. Mode and Frequency

The **mode** is the value or category occurring most often. In the representative Friday, 13 orders used one compatible load and 26 used two. The modal load count is therefore two.

This does not mean every order used two loads. The mean load count is $(13 \times 1 + 26 \times 2)/39 \approx 1.67$. That fractional average is useful for planning across many orders, even though no individual order used 1.67 loads.

There is no useful single mode in the five distinct Wash P/T values above. Several values can tie for most frequent, and rounding continuous durations can create ties. Mode is often more informative for counts or categories, such as load count or correction reason, than for precisely timed work.

### 9.4.6. Keep the Summary Attached to Its Evidence

| Summary | Hand calculation | Question |
|---|---|---|
| Count | Count the relevant rows | How many observations support this? |
| Sum | Add the values | How much across this group? |
| Minimum / maximum / range | Read the ends and subtract minimum from maximum | What extremes and total spread did we observe? |
| Mean | Sum divided by count | What is the average per observation? |
| Median | Find the middle of the sorted list | What is the middle observation? |
| Percentile | Locate a stated position in the sorted list | How did the middle or tail behave? |
| Mode | Tally each value or category | What occurs most often? |

The Mapping Worksheet can preserve the count, sum, and mean needed for the map. The baseline performance readout can add distribution summaries needed for a customer or system question. Both must point back to the observations. Do not treat an empty measurement as zero or average unrelated activities together.

If two groups have different counts, combine their sums and counts before finding the combined mean. For two orders averaging 10 minutes and eight averaging 20 minutes, the combined mean is $(2\times10+8\times20)/10 = 18$ minutes, not 15. The same caution applies to daily promise-violation rates.

> **!** Pause and Check
>
> Using the five Wash observations, calculate count, sum, mean, minimum, maximum, range, and median. Then write one complete measurement statement that identifies the aggregation, quantity, physical unit, observation unit, and group.

### 9.5. From Activity Records to Complete Orders

To build an order row, find its first Drop-off queue entry and its final Stage and notify work end. Subtract the first timestamp from the last. Then group the rows by activity before adding durations across the order. Order-level activities contribute one record each. For Wash and Dry, preserve the separate load records but collapse their parallel timing into the order's elapsed activity span; never add parallel load durations and call the result customer lead time.

Here are WNF-003's activity records in time order, so you can reconstruct an order on paper without opening a data file. All timestamps are elapsed operating minutes; each row is pass 1 of its named activity for this order or load.

| Activity | Load ID | Queue entry | Setup start | Work start | Work end |
|---|---|---|---|---|---|
| Drop-off and ticket | — | 12.1 | 13.4 | 13.4 | 16.6 |
| Sort and tag | — | 16.6 | 16.6 | 16.6 | 21.5 |
| Wash | L1 | 21.5 | 23.9 | 25.1 | 55.1 |
| Wash | L2 | 21.5 | 25.1 | 26.3 | 56.3 |
| Dry | L1 | 56.3 | 56.3 | 57.0 | 108.5 |
| Dry | L2 | 56.3 | 57.0 | 57.6 | 109.1 |
| Fold and package | — | 109.1 | 129.1 | 129.6 | 142.5 |
| Fold correction | — | 142.5 | 148.9 | 148.9 | 153.0 |
| Stage and notify | — | 153.0 | 173.7 | 173.7 | 178.4 |

Subtract work start from work end for P/T, queue entry from setup start for W/T, and setup start from work start for S/T. The resulting order summary begins:

| Order | Arrival | Ready / notified | Total P/T | Total W/T | Total S/T | Operating L/T |
|---|---|---|---|---|---|---|
| WNF-001 | 9.1 | 174.3 | 109.8 | 51.2 | 4.2 | 165.2 |
| WNF-002 | 10.9 | 173.7 | 106.9 | 50.4 | 5.5 | 162.8 |
| WNF-003 | 12.1 | 178.4 | 111.3 | 50.8 | 4.2 | 166.3 |

All values are minutes. For WNF-003, $178.4 - 12.1 = 166.3$, and $111.3 + 50.8 + 4.2 = 166.3$. Its nine activity records include two Wash loads, two Dry loads, and one Fold correction; the other two displayed orders have eight records because they have no correction.

These equalities work because the order summary first collapses each set of parallel Wash or Dry load records into one order-level elapsed span. Do not add parallel machine durations and call the sum customer elapsed time. For a different process, check that property before using the same shortcut.

Counting the two Y flags at Fold and package and Fold correction would double-count WNF-003's defect. One flag identifies the trigger and the other identifies correction work. Count the correction record or the distinct affected order, according to the question.

### 9.5.1. Total Processing Time, Cycle Time, and Lead Time

**Definition 9.1.**

> **Definition: Total Processing Time**
>
> Total processing time is the sum of the processing times on one unit's path, including repeated work. Average total processing time averages those unit totals. Active processing includes automatic machine cycles; it does not mean a worker is occupied throughout.

**Definition 9.2.**

> **Definition: Cycle Time**
>
> Cycle time, written **C/T**, is the observed time between successive completions from a process or process step. For repeating operator work, it can also describe the time required to complete the work elements before the sequence repeats. In a stable, sequential one-piece operation, one unit's P/T and the C/T between completions may be numerically equal; the labels still answer different questions. Cycle time is a completion pace, not the sum of processing times along a unit's path.

The timing-terms note in Chapter 8 identifies the Rother and Shook convention used here and the alternate Hopp and Spearman convention. In either convention, identify the start event, stop event, observation unit, aggregation, and physical unit before comparing two reported numbers.

**Definition 9.3.**

> **Definition: Lead Time**
>
> Lead time, written **L/T**, is the elapsed time between the defined start and stop events for one work unit. Unless another clock is stated, elapsed lead time includes all calendar time within that boundary.

**Definition 9.4.**

> **Definition: Operating Lead Time**
>
> Operating lead time counts only the periods in which the process is scheduled to operate between the defined start and stop events. Wash n' Fold uses drop-off to ready-and-notified and excludes closed hours. Report the operating calendar whenever this measure is used.

For sequential, fully accounted order intervals:

$$\mathrm{L/T} = \mathrm{P/T} + \mathrm{W/T} + \mathrm{S/T}$$

Here P/T includes all active processing, VA and non-VA; W/T includes all waits; S/T includes preparation before processing. Keep setup separate even when it is small. An absent measurement does not justify dropping the S/T term.

### 9.5.2. Check the Worksheet Against the Orders

The activity worksheet mixes 65-load Wash and Dry summaries with 39-order summaries for the other activities. Adding those means would mix observation units and cannot produce order lead time.

Instead, collapse each order first and then average the complete order rows. The order summary gives mean P/T of 104.5 minutes, mean W/T of 554.0 minutes, and mean S/T of 4.1 minutes, which add to the 662.6-minute mean operating lead time. Those order-level means already include the observed correction paths.

If you estimate the correction contribution separately, remember that not every order followed that path. Five of the 39 orders each had one correction record averaging 6.8 minutes of work and 5.6 of waiting.

Their average contribution across **all** orders is:

$$\frac{5}{39}(6.8 + 5.6) \approx 1.6 \text{ minutes per order}$$

Do not add 12.4 minutes to every order or divide correction time by 39 when describing the correction activity itself. The correct denominator depends on the question.

Means can be added when they refer to the same orders and account for how often each path occurs. **Adding activity medians or P90s does not give the median or P90 of complete orders.** Calculate those from the complete order operating-lead-time list.

## 9.6. Typical, Tail, and Promise Performance

An average answers one question about the group. A delivery promise asks whether particular orders met a deadline.

The table below contains all 21 orders dropped off by 9:00 a.m. Before reading further, try to identify the shape of their operating-lead-time distribution, the middle order, and the slow tail by scanning the rows. This is possible, but it is harder than it needs to be.

Use it for the promise analysis; the remaining 18 orders are outside that promise cohort. All timestamps are elapsed operating minutes.

| Order suffix | Arrival | Ready / notified | Operating L/T | Missed promise? |
|---|---|---|---|---|
| 001 | 9.1 | 174.3 | 165.2 | N |
| 002 | 10.9 | 173.7 | 162.8 | N |
| 003 | 12.1 | 178.4 | 166.3 | N |
| 004 | 31.8 | 286.3 | 254.5 | N |
| 005 | 33.7 | 286.9 | 253.2 | N |
| 006 | 44.1 | 290.5 | 246.4 | N |
| 007 | 47.8 | 390.9 | 343.1 | N |
| 008 | 52.8 | 391.9 | 339.1 | N |
| 009 | 60.3 | 394.6 | 334.3 | N |
| 010 | 61.2 | 485.7 | 424.5 | N |
| 011 | 62.1 | 485.3 | 423.2 | N |
| 012 | 63.6 | 488.8 | 425.2 | N |
| 013 | 75.3 | 593.4 | 518.1 | N |
| 014 | 78.2 | 589.7 | 511.5 | N |
| 015 | 81.0 | 590.5 | 509.5 | N |
| 016 | 88.0 | 686.3 | 598.3 | Y |
| 017 | 89.7 | 685.7 | 596.0 | Y |
| 018 | 97.4 | 688.8 | 591.4 | Y |
| 019 | 100.0 | 782.0 | 682.0 | Y |
| 020 | 103.1 | 781.5 | 678.4 | Y |
| 021 | 110.5 | 786.7 | 676.2 | Y |

Now inspect the same orders as points arranged from fastest to slowest.

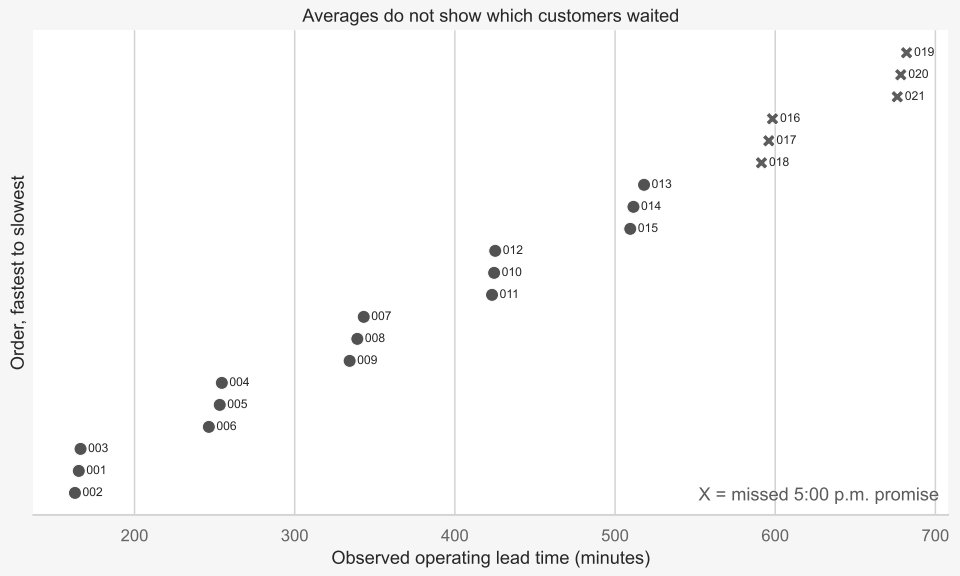



**Figure 9.2.:** The 21 promise-eligible Wash n' Fold orders arranged by observed operating lead time.

The groups of similar operating lead times, the gaps between them, and the six promise misses are visible without asking the eye to coordinate five table columns. The table remains necessary when we need an exact timestamp or must reproduce a calculation. The plot and table answer different questions about the same evidence.

The 21 operating lead times sum to 8,899.2 minutes, so their mean is $8,899.2/21 \approx 423.8$ minutes. Sorted, they are:

162.8, 165.2, 166.3, 246.4, 253.2, 254.5, 334.3, 339.1, 343.1, 423.2, **424.5**, 425.2, 509.5, 511.5, 518.1, 591.4, 596.0, 598.3, **676.2**, 678.4, 682.0.

The minimum is 162.8, the maximum is 682.0, and the range is $682.0 - 162.8 = 519.2$ minutes. Using the nearest-rank convention, $Q_1$ is the sixth observation, 254.5 minutes; the median is the 11th observation, 424.5 minutes; and $Q_3$ is the 16th observation, 591.4 minutes.

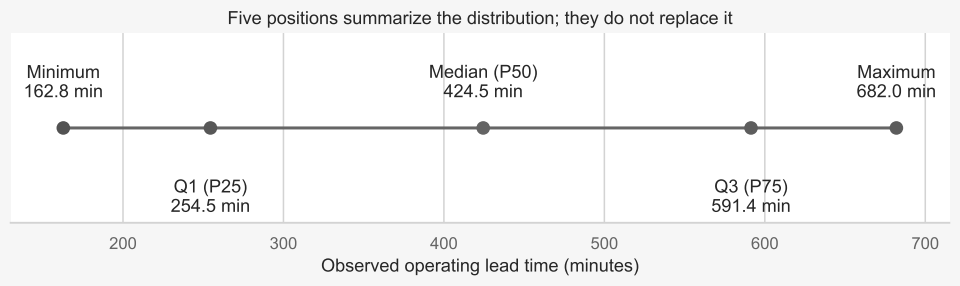



**Figure 9.3.:** The five-number summary marks five positions in the eligible-order distribution.

The mean of 423.8 minutes happens to sit near the median, but that agreement does not make the distribution narrow. Half of the observed lead times span more than 336 minutes between $Q_1$ and $Q_3$.

### 9.6.1. Find P90 by Hand

The **90th percentile**, or **P90**, is a way to describe the slow tail. For this chapter, use the **nearest-rank** rule:

1. Sort the observations from smallest to largest.
2. Multiply the count by 0.90.
3. Round that position upward to a whole number, if necessary.
4. Read the observation at that position, counting from one.

For 21 orders, $0.90 \times 21 = 18.9$, so use position 19. P90 is therefore **676.2 minutes**: at least 90% of these observed orders had operating lead times no

greater than this value. It is a sample description, not a guarantee about the next order.

An **empirical cumulative distribution**, or ECDF, plots every observed value against the proportion of observations at or below it. To read a percentile, begin with a proportion on the vertical axis, move to the observed staircase, and then read the corresponding time on the horizontal axis.

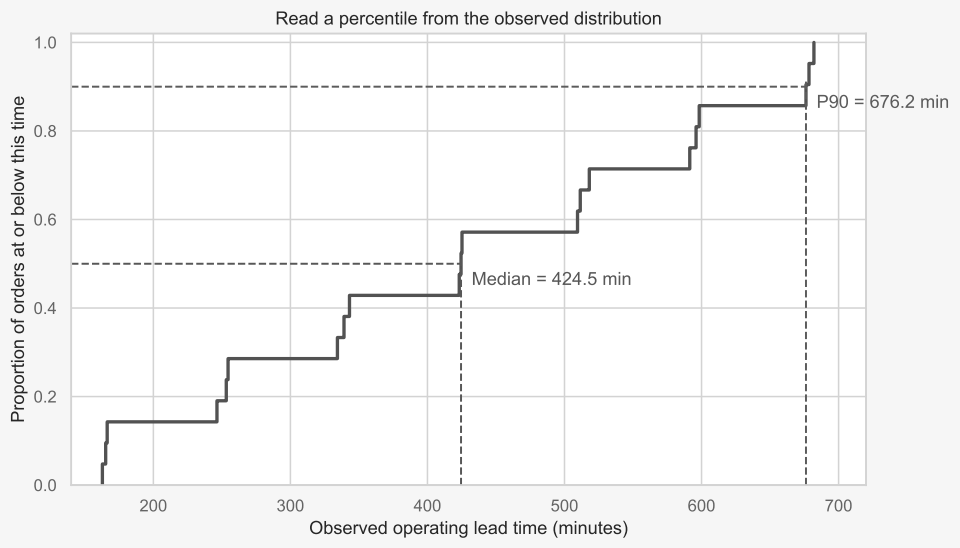



**Figure 9.4.:** The empirical cumulative distribution connects observed operating lead times to percentiles.

The ECDF preserves all 21 observations while making P50 and P90 visible. Unlike a histogram, it does not require us to choose arbitrary bin widths.

Software offers several percentile conventions, including interpolation between observations. Record which one you use. The technical simulation exports use an interpolated P90; the pencil-and-paper readout here uses nearest rank consistently. A small difference between those results is a method difference, not a change in the orders.

### 9.6.2. Count Promise Violations

An order is eligible if arrival is at or before minute 120. An eligible order violates the promise if readiness notification is after minute 600. Exactly 5:00 p.m. counts as on time.

Let $M$ be the number of eligible orders that miss the promise and $E$ be the total number of eligible orders.

$$\begin{aligned}\text{Violation rate} &= \frac{M}{E} \times 100\% \\ &= \frac{6}{21} \times 100\% \\ &\approx 28.6\%.\end{aligned}$$

WNF-016 has a lead time of 598.3 minutes, which is below 600, but it still misses the promise. It arrived at minute 88.0 and was ready at 686.3. A fixed clock deadline is different from a fixed allowance measured from arrival.

The mean and P90 describe operating lead times; the violation rate tests the promised deadline. None substitutes for the others.

### 9.6.3. Keep the Two Cohorts Separate

| Measure | All 39 orders | 21 promise-eligible orders |
|---|---|---|
| Mean operating L/T (min) | 662.6 | 423.8 |
| Median operating L/T (min) | 678.4 | 424.5 |
| Minimum / maximum (min) | 162.8 / 1,152.8 | 162.8 / 682.0 |
| P90, nearest rank (min) | 1,057.8 | 676.2 |
| Promise-violation rate | Not defined for this whole group | 6/21 = 28.6% |

Later arrivals also experience the growing release queue. Including them changes the operating-lead-time summary; it does not change which orders qualify for the 9:00 a.m. promise. The project target remains approximately 28% to 10% within 30 days.

The baseline performance readout should preserve this distinction rather than offering one unlabeled “current average.” For each reported metric, record the quantity, value, unit, aggregation, observation unit, boundary, cohort or period, and any calculation convention that another reader would need to reproduce it.

> **!** Compare Two Questions
>
> Explain why a report of “average operating lead time 423.8 minutes” is incomplete without its cohort. Then explain why dividing the six violations by all 39 orders would misrepresent promise performance.

### 9.7. Read the Process as a System

The distribution tells us what customers experienced. It does not yet tell us whether the shop can keep pace with demand, where elapsed time accumulates, or how unfinished work relates to completion rate. The next measures answer those system questions.

### 9.8. Takt Time, Demand, and Capacity

The first system question is whether the process can finish work at the pace customers require. Takt translates demand into that required pace; capacity estimates what the available resources could accomplish; throughput records what the process actually completed.

**Definition 9.5.**

> **Definition: Takt Time**
>
> Takt time is available production time in a planning period divided by demand for that same period:
>
> $$T = \frac{A}{D}$$

For planning, use approximately 38 orders per Friday and 690 operating minutes:

$$T = 690/38 \approx 18.2 \text{ minutes per order}$$

Using the simulation's unrounded average demand of 37.73 gives 18.3 minutes. Using the representative day's actual 39 orders gives 17.7 minutes. Label whether the calculation uses forecast demand or an observed day's demand.

The cell has two workers, six washer positions, six dryer positions, and two folding stations. Do not multiply the shop's 690-minute clock by its employee count for customer Takt. Staff minutes matter separately when estimating labor capacity. The model assumes these resources are available for the stated operating period; in an observed business, scheduled downtime and staffing changes need explicit treatment.

Orders arrive between 7:00 and 11:00, much faster than the whole-day completion pace. The modeled mean interarrival time is 6.4 minutes. Arrival

cadence, required average completion pace, and actual throughput are three different measures.

A 45-minute dry cycle does not by itself imply capacity of one order every 45 minutes. Several orders can dry in parallel, and some occupy two positions. A supplied planning estimate using the current modeled work content is:

| Resource | Estimated capacity (orders/hour) |
|---|---|
| Two workers, across all their tasks | 3.56 |
| Six washer positions | 6.28 |
| Six dryer positions | 4.46 |
| Two folding stations | 6.69 |

These estimates use the 100-run current-state work mix and include setup and correction work. For example, approximately 33.7 worker-minutes per order gives $2 \times 60/33.7 \approx 3.56$ orders/hour. They are resource-workload estimates, not observed completion rates or guaranteed schedules.

The smallest estimate is 3.56 orders/hour. Daily demand averages about 3.28 orders per operating hour, below that physical estimate. This justifies testing whether the release rule leaves usable capacity idle. It does not guarantee that a burst of morning orders will meet a 5:00 p.m. deadline. The log is needed to check what the process actually accomplishes.

## 9.9. Process Cycle Efficiency

The next question is where the customer's elapsed time goes. If active work occupies only a small part of lead time, making one active step slightly faster may leave the customer experience almost unchanged.

**Definition 9.6.**

> **Definition: Process Cycle Efficiency**
>
> Process cycle efficiency, or PCE, is the share of lead time spent in value-added work:
>
> $$PCE = \frac{\text{Value-added time}}{\text{Lead time}} \times 100\%$$
>
> For a group of orders, this book's White Belt convention is to divide total VA time by total lead time for the same orders. That equals mean VA time divided by mean lead time for that group. It need not equal an unweighted average of individual orders' PCE percentages. When the study uses an operating clock, name the denominator as operating lead time and use that same clock throughout.

Wash, Dry, and Fold and package supply the VA time in this case. Correction, intake, sorting, staging, waiting, and setup remain outside the numerator.

Across the 39 orders, VA time sums to 3,582.3 minutes and operating lead time sums to 25,840.5 minutes:

$$PCE = \frac{3,582.3}{25,840.5} \times 100\% \approx 13.9\%$$

This says that approximately 13.9% of the summed order operating lead time is classified VA. It is not worker utilization, machine utilization, or a universal quality score. There is no universal "good PCE" threshold in this chapter. Compare like boundaries and ask which non-VA contributions can be reduced without harming quality.

The order-level P/T, W/T, and S/T means reveal the practical consequence more directly.

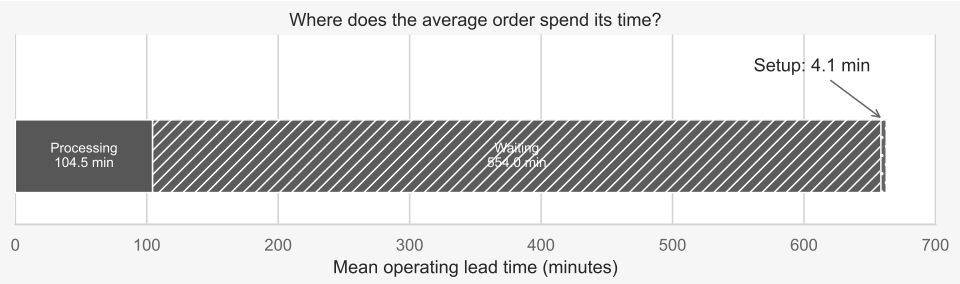



**Figure 9.5.:** Waiting accounts for most of the mean Wash n' Fold operating lead time.

Waiting accounts for 554.0 of the 662.6 mean operating minutes. That does not prove why the orders waited, but it tells the team why shaving a minute from an already short active task is unlikely to solve the customer problem.

## 9.10. WIP, Throughput, and Little's Law

The final system question connects unfinished work, completion rate, and elapsed time. This matters because a team can move work out of sight, count it at a convenient instant, or celebrate a busy resource without improving the rate at which complete orders leave the system.

Work in process, or WIP, is unfinished work inside the chosen boundary. Throughput is completions per unit of time. Little's Law relates averages:

$$\begin{aligned}\text{Average WIP} &= \text{Throughput} \\ &\quad \times \text{Average lead time}\end{aligned}$$

$$\text{Average lead time} = \frac{\text{Average WIP}}{\text{Throughput}}$$

Use the same work unit, boundary, and time basis throughout. Average WIP describes how much work is present over the observation period. It is not the number present at one convenient instant.

### 9.10.1. A Small WIP Calculation

Suppose there are two open orders for 30 minutes and four for the next 30 minutes. Average WIP over that hour is $(2 \times 30 + 4 \times 30)/60 = 3$ orders.

For unequal durations, weight each count by how long it persisted. Two orders for 10 minutes and four for 50 minutes gives $(2\times10+4\times50)/60 \approx 3.67$, not three.

### 9.10.2. Use the Relationship, Then Check Its Assumptions

Suppose a support process averages 50 open tickets over a representative period and completes 10 tickets per working day:

$$\begin{aligned}\text{Average lead time} &= \frac{50 \text{ tickets}}{10 \text{ tickets/day}} \\ &= 5 \text{ working days}\end{aligned}$$

At unchanged throughput, a three-day target corresponds to $10 \times 3 = 30$ average open tickets. That relationship does not tell us which operational change will achieve it. Simply moving tickets outside the reported boundary would improve the number without improving customer experience. Reducing WIP too far can also starve resources and reduce throughput.

### 9.10.3. Reconcile the Wash n' Fold Full Run

The representative run starts empty at minute zero and follows all 39 orders to completion at approximately minute 1,377.6. For this full cohort, the sum of order operating lead times is also the area under the open-order WIP history: 25,840.5 order-minutes.

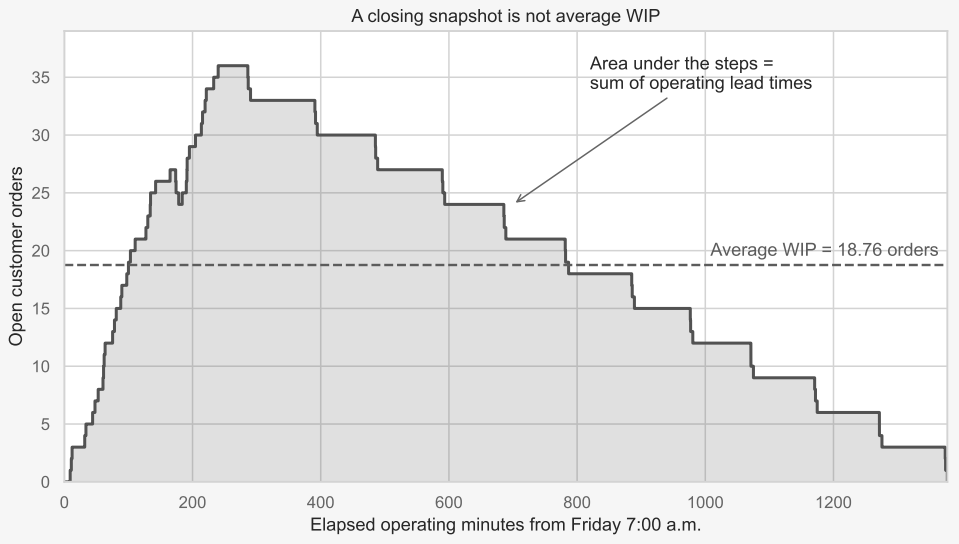



**Figure 9.6.:** Open-order WIP changes throughout the representative Friday and its completion tail.

The chart makes the distinction between a snapshot and an average visible. The process has 21 open orders at closing time, but the height of the curve changes throughout the run. Average WIP depends on the whole shaded area, not one selected point. Dividing 25,840.5 order-minutes by the 1,377.6-minute run gives:

$$\text{Average WIP} \approx 18.76 \text{ orders}$$

Mean operating lead time is $662.6/60 \approx 11.04$ operating hours. Little's Law therefore gives approximately:

$$\text{Throughput} = \frac{18.76}{11.04} \approx 1.70 \text{ orders/hour}$$

The direct count agrees: $39/(1,377.6/60) \approx 1.70$ orders/hour. This is an accounting check over a complete start-empty/end-empty run, not independent proof of the simulation or a claim that Friday arrivals were steady. For ongoing operations, use a representative period and account for work carried across its boundaries.

At the 6:30 p.m. close, only 18 orders are ready and 21 remain open. That snapshot is useful operationally, but pairing 21 with the full-run mean operating lead time would mix different measures. Likewise, $18/11.5 \approx 1.57$ orders/hour describes by-close throughput, not full-run throughput.

The capacity estimate of 3.56 orders/hour and the realized full-run rate of 1.70 answer different questions. Their gap supports investigating the serial rule that holds the next wash batch until all orders in the active batch clear drying. Little's Law relates the observed averages; capacity and release rules help explain why they take those values.

## 9.11. Build the Baseline Performance Readout

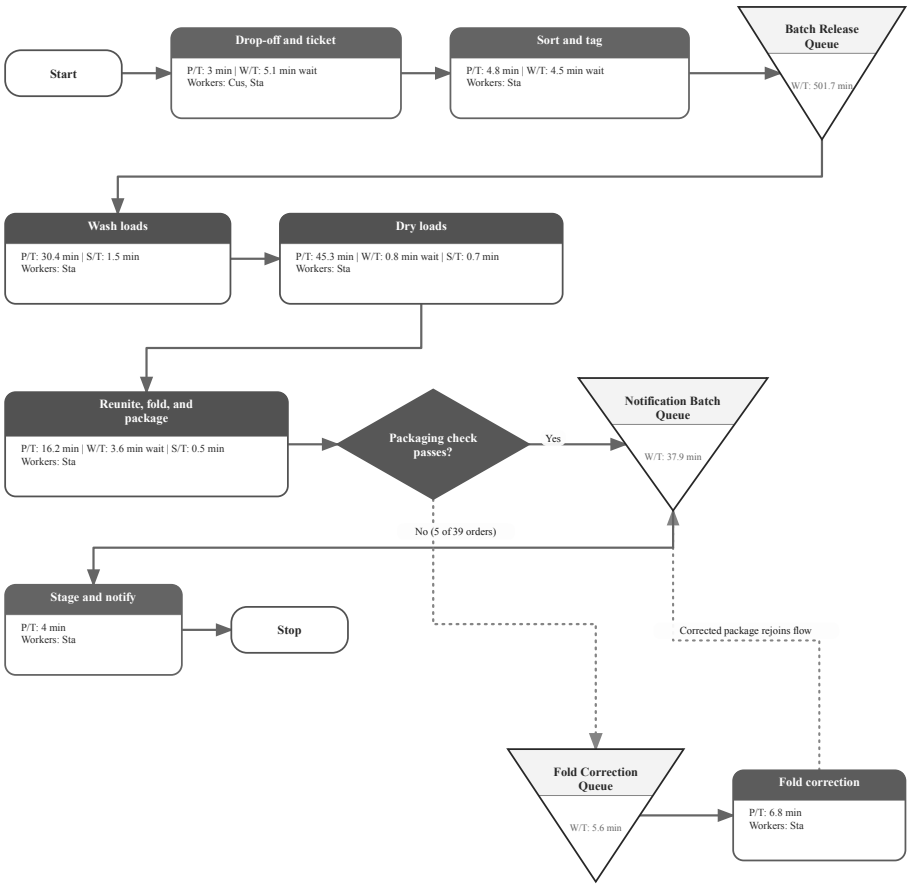



**Figure 9.7.:** Wash n' Fold current-state map; timing labels state whether their evidence comes from 39 orders, 65 loads, or five correction records.

The map shows where work moves and waits. The **Baseline Performance Readout** records what the selected measures say about the defined process and what the team should inspect next.

For every reported metric, the readout names:

- the quantity and value,
- whether it describes one observation or an aggregate,
- the observation unit and physical unit,
- the process boundary,
- the cohort and observation period, and
- the operational question or next check.

Wash n' Fold uses the completed readout below as the model. For your capstone, construct the same kind of compact readout from your before-state observations. Use the blank **Baseline Performance Readout** as a PDF, XLSX, or CSV. The completed Friday 91 version is available as a PDF, XLSX, or CSV. Its separate metric rows keep the value, aggregation, units, boundary, cohort, clock, supported signal, and next question inspectable. The charts in this chapter help you learn to inspect distributions; creating those charts is not a White Belt capstone requirement.

**Readout scope:** simulated representative Friday 91; drop-off through staging and readiness notification; operating minutes from 7:00 a.m.; one customer order as the case unit. Rows identify whether they use all 39 orders, the 21 promise-eligible orders, or a smaller set of activity records.

| Measure and result | Aggregation and unit | Supported signal | Next question or check |
|---|---|---|---|
| Mean operating L/T 662.6 min; mean W/T 554.0 min | Means across all 39 orders; minutes per order | Waiting dominates elapsed operating time | How does batch release interact with available resources? |
| Batch Release Avg W/T 501.7 min | Mean across 39 order-level waits; minutes per order | The biggest delay is before Wash | Observe idle washer positions while orders await release |
| Median operating L/T 424.5 min; P90 operating L/T 676.2 min | Distribution of 21 promise-eligible orders; minutes per order | The middle hides a much slower tail | Trace later eligible arrivals through the release queue |
| 6 of 21 eligible orders miss 5:00 p.m. (28.6%) | Count and rate among 21 eligible orders; orders and percent | The customer promise is unreliable on this Friday | Check comparable Fridays before generalizing |

| Measure and result | Aggregation and unit | Supported signal | Next question or check |
|---|---|---|---|
| Five correction records in five orders | Occurrences and affected orders among all 39 orders | Correction is evidenced and adds NVA time | Investigate the recorded packaging-check trigger |
| PCE 13.9% | Ratio of total VA time to total lead time across all 39 orders; percent | Most elapsed time does not transform the order | Identify which measured delay offers the strongest improvement target |
| Mean VA time 91.9 min; mean time outside VA 570.7 min | Means across all 39 orders; minutes per order; time outside VA includes RNVA and NVA | Active transformation occupies a small share of the order's elapsed time | Keep necessary support work distinct from avoidable loss when diagnosing that remainder |
| Realized throughput 1.70 orders/hour; physical planning estimate 3.56 orders/hour | Full-run completion rate compared with a capacity estimate; orders per operating hour | Available physical capacity may not be reaching customers as completed orders | Test the release-rule hypothesis using the same demand and resources |

A metric symptom supports a question; it does not establish every cause or authorize a solution. Chapter 10 begins with the readout's supported signals and next checks, uses the Eight Wastes to classify the observed losses, and distinguishes evidence from competing explanations. Carry the readout and verified map into that diagnosis. For your own process, retain the same blank readout structure for the after-state comparison so a changed denominator or boundary cannot masquerade as an improvement.

## 9.12. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Apply) A registrar's office has ten eight-hour working days to handle 400 requests. Calculate Takt in minutes per request. One dedicated worker needs 11 minutes per request on average. What does that suggest, and what assumptions should you check?

**2.** (Apply) A mortgage process has 15 minutes of RNVA intake, two workdays waiting, 45 minutes of VA review, three workdays waiting, 90 minutes of VA underwriting, and 10 minutes of RNVA notification. Use eight-hour workdays. Calculate lead time and PCE. If VA time stays unchanged, what maximum lead time corresponds to a 20% PCE target?

---

**3.** (Apply) A repair shop averages 24 open repairs and completes six per working day. Use Little's Law to calculate average lead time. If throughput remains six per day, what average WIP corresponds to a three-day target? Explain why moving unfinished repairs into an uncounted storage area does not achieve the customer goal.

---

**4.** (Apply) Five activity P/T observations are 4, 6, 6, 9, and 15 minutes. Calculate count, sum, minimum, maximum, range, mean, median, mode, and the nearest-rank five-number summary. A second group contains 15 observations with a sum of 210 minutes. Calculate the combined mean.

---

**5.** (Apply) Reconstruct WNF-003's order row from its event-log records. Show the arrival-to-notification calculation and the P/T + W/T + S/T check. Count its activity records, named activities, correction records, and distinct affected orders. Explain why a worksheet row for Fold correction uses a different denominator from mean correction time per order.

---

**6.** (Analyze) Inspect Figure 9.1. Explain why the matching averages do not justify treating its four distributions as equivalent. Then compare these two ten-order cohorts, whose lead times are in minutes:

A: 10, 10, 10, 10, 10, 10, 10, 10, 10, 10.

B: 5, 5, 5, 5, 5, 5, 5, 5, 25, 35.

All have the same promise of completion within 20 minutes of arrival. Compare mean, median, nearest-rank P90, and promise-violation rate. What does the shared mean conceal?

---

**7.** (Analyze) Using the 21-row Wash n' Fold table, verify the mean, median, and nearest-rank P90. Then use Figure 9.2 and Figure 9.4 to describe one pattern that is difficult to see in the table alone. Check eligibility and count promise violations. Explain why WNF-016 is late despite a lead time shorter than 600 minutes.

---

**8.** (Analyze) The Wash n' Fold Batch Release mean wait is 501.7 minutes. The worksheet includes the same waiting interval that the map promotes to a queue triangle. Explain why adding both would be wrong. Name one supported symptom, one release-policy hypothesis, and the next observation needed.

---

**9.** (Analyze) A team removes intake and the release queue from its reported lead time but retains the full process's VA numerator. Its PCE rises. Explain the boundary error and describe a fair comparison.

---

**10.** (Apply) A stable application service averages 12 open applications and an end-to-end lead time of three working days. Estimate completions per day, choosing the appropriate relationship yourself. Could a closing snapshot of 12 applications replace the measured average? Explain.

---

**11.** (Analyze) A colleague adds the Wash median, Dry median, and Fold median and calls the result median process lead time. Identify two problems. Explain which records would support the correct calculation.

---

**12.** (Analyze) Write a capstone-ready statement using the representative Friday. Name the cohort, boundary, one metric symptom, one waste hypothesis, and the next evidence needed. Keep simulated observations, planning estimates, and causal claims distinct.

### 9.12.1. Journal and Reflect

Describe how your judgment changed when you compared two groups with the same mean but different missed promises. Which measure would you now ask for before reassuring a customer?

Recall a metric you have seen reported without a clear denominator or boundary. What would you ask its owner, and how would you explain any resulting correction without blaming them?

## 9.13. Chapter Summary

- The event log preserves individual activity records; the worksheet and map summarize activities; the order table preserves each customer's complete experience; and the Baseline Performance Readout records decision-relevant measures and next questions.
- A distribution describes the observed values and where they occur; tables, dot plots, and ECDFs expose different features of the same evidence.
- Count, sum, minimum, maximum, range, mean, median, percentiles, and mode answer different questions about a named group.
- The conventional five-number summary is minimum, $Q_1$, median, $Q_3$, and maximum.
- Means require counts and compatible groups; branch work requires frequency weighting, and activity medians or P90s do not add into process medians or P90s.
- P90 describes the observed slow tail; the promise-violation rate directly tests the promised deadline for eligible orders.
- Takt states the required demand pace, capacity estimates describe available resources, and throughput records actual completions.
- PCE and Little's Law require matching boundaries and units; average WIP is different from a closing snapshot.
- The representative Friday supports investigating batch release, while causal conclusions need further evidence.

# 10.

# The Eight Wastes

> "The goal is not to speed up the process, but to eliminate the time when nothing is happening." — Shigeo Shingo

## 10.1. Why This Chapter Matters

A process can look productive while quietly burning time, money, and morale. People stay busy, but work still stalls, rework grows, and customers wait longer than expected. The Eight Wastes framework gives you a practical vocabulary for spotting these losses quickly. Naming a waste gives people a shared way to recognize and discuss it. Removing it still requires finding out why it occurs and testing a change.

## 10.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Lean Six Sigma Principles and Tools Identify and distinguish the Eight Wastes in service and production processes **Understand**
- Lean Six Sigma Principles and Tools Diagnose observed waste, supported inference, and unevidenced hypotheses using current-state map evidence, flow metrics, P90 tail behavior, and customer-promise performance **Analyze**
- Lean Six Sigma Principles and Tools Frame an evidence-linked waste diagnosis that names competing explanations and the next observation needed to distinguish them **Analyze**

## 10.3. Wash n' Fold Project: Do

The Wash n' Fold Baseline Performance Readout now shows five correction occurrences and a long release queue before Wash. Keep the completed readout and verified current-state map beside the diagnosis. They tell us where work stalls or circles back, but we still need to distinguish an observed loss from a suspected cause. The Eight Wastes give us names for those losses and help us decide what to investigate before designing a change.

In Chapter 8, we classified steps as value-added, required non-value-added, or non-value-added. Here we use the map and measurements to make a more specific diagnosis: what kind of waste appears, what evidence supports it, and what should change if a countermeasure works?

### 10.3.1. The Eight Wastes

The SPPM's value-classification rule still applies: keep value-added work, make required non-value-added work more efficient, and investigate eliminating work that is neither valuable nor required.

What we want to identify now are eight different kinds of waste. Seven of these were introduced in (Ohno 1988). These are not the only possible wastes, but they are common enough to be worth memorizing. The acronym DOWNTIME helps:

- Defects
- Overproduction
- Waiting
- Non-Utilized Talent
- Transportation
- Inventory
- Motion
- Extra Processing

For each waste, we will start with an ordinary example and then return to Wash n' Fold. Where a principle helps us distinguish the problem or choose a test, we will use it.

#### 10.3.1.1. Defects

**Definition 10.1.**

> **Definition: Defect**
>
> A defect is a failure of a product or service to fulfill a stated requirement or a reasonable expectation for use, including applicable safety requirements.

A pharmacy fills a prescription with the wrong quantity and must stop delivery, correct the package, and check it again. The correction uses time without producing another unit of customer value.

Unclear customer instructions, unstable work methods, inadequate training, worn equipment, and weak handoffs can all produce defects. Check the actual failure reason before choosing a response: clarify the order, improve the handoff, maintain equipment, or standardize a method that works. Count defects, affected units, and correction time separately, and check that the change improves first-pass yield.

**Definition 10.2.**

> **Definition: First-Pass Yield**
>
> First-pass yield, or FPY, is the proportion of units that complete a defined step or process without a defect, correction, scrap, or rework on their first pass:
>
> $$\text{FPY} = \frac{\text{Units completed correctly on the first pass}}{\text{Units entering the defined boundary}}$$
>
> State the process boundary, observation period, and denominator. One unit can contain more than one defect, so FPY is not the same as a defect count.

> Note
>
> Sometimes quality checks can be required, for instance, when they are required by a contract or applicable regulation. But even so, they are still not adding value to the product since they would be unnecessary if the work were done correctly the first time.

At White Belt level, the key diagnostic question is simple: where could one bad tag or one bad handoff force the work to circle backward?

Five orders in the Wash n' Fold log fail the packaging check and return for Fold correction; the map shows that return branch. The records establish rework, not whether bad tags, mixed items, or missed instructions caused it.

Correction can increase lead time and reduce PCE, but the next observation should identify the failure reason before selecting a countermeasure.

#### 10.3.1.2. Overproduction

Overproduction means making more than the customer needs now, or releasing more work than the next step can absorb.

A print shop runs 500 flyers for an event whose organizer ordered 200, or prints all of tomorrow's jobs before the finishing station has room for them. The first case makes unwanted output; the second creates a pile of work in progress.

Pull authorizes production from actual demand and downstream capacity; keeping a machine busy is not sufficient authorization.

Large-batch incentives, forecasts treated as firm orders, and schedules that reward keeping a machine busy can release too much work. Compare release quantities with demand and downstream occupancy, then test smaller releases or a visible WIP limit. Do not replace a restrictive rule with unrestricted release: that may only move the queue.

The Wash n' Fold case does not show finished laundry made for imaginary customers or too much work released downstream. Its serial rule holds the next wash batch until every order in the active batch clears drying, even when some machine positions are available. The large queue before Wash is waiting; it does not by itself prove overproduction. Inspect release times, available positions, and downstream occupancy before applying this label.

#### 10.3.1.3. Waiting

Waiting is time when a unit of work is ready for a next step but cannot move forward. It occurs in manufacturing and service work alike.

A completed permit application sits in an inbox for two days before the next reviewer opens it. The applicant experiences those two days even if reviewing the form takes only minutes.

Batch release rules, uneven arrivals, missing information, and mismatched handoffs can make work wait. Measure the queue by step before changing a rule; then test smaller batches, clearer handoffs, or a controlled release signal. Check both the local wait and the customer's end-to-end lead time so a reduction in one queue is not merely a transfer to another.

The Wash n' Fold map measures a 501.7-minute mean release wait before Wash, compared with 3.6 minutes before Fold and package; other waits occur

before Dry and notification. This is the clearest observed waste in the case. Chapter 9's resource-workload estimate suggests enough physical capacity for average daily demand, yet a morning arrival burst and the serial release rule can still leave orders waiting. Observe queued orders and idle positions together, then ask what would happen downstream if release changed.

#### 10.3.1.4. Non-Utilized Talent

Non-utilized talent occurs when people's knowledge, skills, or judgment could improve the work but the organization fails to use them.

A clinic receptionist notices that patients repeatedly arrive without a required form and proposes a clearer reminder, but nobody with authority hears or tests the suggestion. The problem persists even though someone close to the work sees it.

Mapping the process requires learning how people actually do the work, not only how a procedure says they do it.

Narrow decision authority, a suggestion box with no follow-up, and work targets that leave no time to improve the method can suppress useful knowledge. Ask operators what they observe, test promising suggestions with them, and report what happened. Idle time establishes waiting; it does not by itself establish unused talent.

In Wash n' Fold, ask whether operators have noticed the effects of the release rule and whether their suggestions are considered. The timing log cannot answer that question. Trace what happens to a concrete suggestion before diagnosing this waste.

#### 10.3.1.5. Transportation

Transportation waste is unnecessary movement of materials or information between places.

A repair shop carries each finished part across the building for inspection, then carries it back to the same bench for packing. Neither trip changes the part in a way the customer values.

The named activities alone can hide a route that sends work back and forth across a room; the process map should make that route visible.

A distant layout, separated handoff points, and poorly placed storage can create extra trips. Trace the route and count trips before moving stations or changing a handoff; compare travel time and end-to-end performance afterward.

In Wash n' Fold, laundry must travel from intake through sorting, washing, drying, folding, and staging. Excessive travel is possible, but the current records contain no station distances or trip counts. Observe the physical route before proposing a layout change or claiming a transportation saving.

#### 10.3.1.6. Inventory

Inventory waste is inventory in excess of what the process needs, including unfinished work between steps. Some supplies and buffers are necessary; the question is whether the amount helps or obstructs delivery.

A repair desk accepts dozens of devices into an open pile while its technicians can complete only a few each day. The pile takes space, hides priority, and makes customers wait longer.

Queues and open-order counts show where unfinished work accumulates. A pull rule limits new releases to what downstream steps can handle.

Large releases, uncertain priorities, and bottlenecks can produce excess WIP. Track open work over time and its location, then test a visible limit or smaller release with a way to keep usable capacity supplied. Check lead time and completion as well as WIP: a smaller pile alone is not proof of better service.[1]

In Wash n' Fold, the inventory of interest is unfinished orders awaiting release, partly processed loads, and orders awaiting notification, not just detergent on a shelf. The simulated Friday has 21 unfinished orders at close. That snapshot establishes open work, but the amount to reduce and the right place to limit it require a flow explanation. Finished orders awaiting pickup occupy space, although pickup lies outside the measured boundary.

#### 10.3.1.7. Motion

Motion waste is unnecessary movement by the person doing the work, such as searching, reaching, or walking for a needed tool. Transportation concerns movement of the work or materials; motion concerns movement of the worker.

A laboratory technician walks to a distant cabinet for a label after preparing every sample because labels have no assigned place at the bench. Each sample triggers the same avoidable trip.

[1] Inventory can appear as an asset on a balance sheet, but a larger balance does not by itself mean better performance. Measures such as inventory turnover ask how quickly a business converts inventory into sales; a process change still needs to be judged by its actual costs and results.

An observer may have to look inside a named step to see repeated searches and trips that the map alone conceals.

Poor workplace organization, misplaced supplies, and clutter often produce motion waste. Put frequently needed items at the point of use, give them clear locations, and measure retrieval time before and after. The next chapter develops 5S as one way to make that arrangement reliable.

The Wash n' Fold setup method uses one shared supply jug that must be found, fetched, measured, and passed among washer positions. That context makes avoidable motion plausible, but the log does not separately time each search or trip. Observe setup to estimate the local saving, then check lead time and promise performance; PCE alone cannot establish which customer outcome improved.

#### 10.3.1.8. Extra Processing

Extra processing is work beyond what is needed to meet the customer's need and any legitimate process requirement.

A team copies the same address into three systems and prints a paper form that no one uses. The extra entries and printout do not improve the customer's result.

Duplicate checks, unclear standards, and compensating for earlier defects can make extra work look responsible. People often justify an added step as being "careful." Sometimes it is careful; sometimes it compensates for a weak upstream process. Ask what each step accomplishes before removing it; test a simpler method while monitoring defects, lead time, and customer outcomes. A shorter check is no improvement if it lets an incorrect order through.

At Wash n' Fold, duplicate checking, re-tagging, repeated handling, and unnecessary sorting are possibilities to investigate, not recorded findings. Observe their purpose and frequency before calling them extra processing. The recorded Fold correction is already defect-related rework; do not count it again as a separate measured loss.

With all eight categories in view, ask two questions of the Wash n' Fold map:

1. Where would this waste appear on the Wash n' Fold map?
2. What local measure would test whether we reduced it, and what customer outcome would tell us whether that helped?

### 10.3.2. Wash n' Fold Waste Summary

The current-state map tells us where to look first. The mean release wait before Wash is 501.7 minutes, compared with 3.6 before Fold and package. Among the 21 promise-eligible orders, median operating lead time is 424.5 minutes, P90 is 676.2, and six orders miss 5:00 p.m. The long tail matters because a reasonable-looking middle does not keep the promise for those six customers.

The table summarizes the first-pass diagnosis. An observed condition is a place to investigate, not automatic proof that the proposed cause or countermeasure is right. Some categories remain unestablished because this simulated Friday does not record the observations needed to assess them; there is no need to fill every box with a claim.

| Waste | Evidence status in this simulated Friday | First useful check |
|---|---|---|
| Defects | Observed: five packaging-check failures lead to Fold correction | Correction count and reason |
| Overproduction | Not established: the serial release rule holds work before Wash; a queue alone does not prove excessive upstream production | Release times, available positions, and downstream occupancy |
| Waiting | Observed: mean release wait 501.7 min; notification wait 37.9 min | These waits, then eligible-order P90 and promise violations |
| Non-Utilized Talent | Hypothesis: operator knowledge may be excluded from release decisions | Ask operators what they would change and why |
| Transportation | Not evidenced as excessive: movement between stations is necessary, but distances are not recorded | Trace the physical route before proposing a layout change |
| Inventory | Observed: 21 unfinished orders at close; whether and how to reduce it requires a flow explanation | Open-order counts over time, not just one snapshot |
| Motion | Setup context identifies a shared supply jug; avoidable searching is a hypothesis, not a timed finding | Observe fetching and measuring within setup |

| Waste | Evidence status in this simulated Friday | First useful check |
|---|---|---|
| Extra Processing | Not evidenced separately: do not invent duplicate checks or count correction twice | Observe whether checks exceed customer or process requirements |

### The Least Cost Way!

In my experience, people's first thought about improvement is often to make a value-added step faster, perhaps by buying an expensive machine. At Wash n' Fold, faster washers would be an attractive purchase, but they would not release orders that are waiting for permission to enter the machines already available. The map and baseline give us a better place to start investigating. This is the Least Cost Way: examine the waste before committing to equipment or staffing changes, while accounting for the cost, safety, and human consequences of any proposed change.

### Case Prompt: Choose the First Wash n' Fold Tests

Using the table and your current SPPM, choose two wastes to investigate first. For each, identify the observed evidence, the cause that remains uncertain, a bounded next observation or low-cost test, and both a local measure and a customer-outcome measure. Explain why these two deserve attention before an equipment purchase. You may leave other categories unestablished.
Record your choices in your own notes or course workbook before opening the reference response.

### Reference Response: First Wash n' Fold Tests

Waiting before Wash is a strong first choice: its mean release delay is 501.7 minutes, and the serial rule is a plausible cause to test rather than a cause proved by the average alone. Observe release decisions, queued orders, and available washer and dryer positions; then test a controlled release change with a WIP limit. Track the release wait locally and eligible-order P90 and 5:00 p.m. promise violations for customers.
Defect-related correction is another defensible choice because five orders return from the packaging check. Record correction reasons before changing the handoff or work method, then track FPY, correc-

> tion time, and operating lead time to ready-and-notified. Open WIP also deserves attention, but do not count the same held orders as an independent cause of their measured wait. The case does not yet establish excessive transportation, extra processing, unused talent, or overproduction.

### 10.3.3. Transfer Practice: Breakfast Service

The following breakfast-counter scenario is fictional.[2] Read it as you would an observation of an unfamiliar process: identify what it establishes, keep uncertain causes open, and notice the safety concern.

A customer orders one slice of wholemeal toast, uncut, with butter. The cashier writes the order on a slip and places it in the preparation rack behind another ticket. Only "one toast, butter" is visible to the worker at the toaster. The worker selects one slice of white bread, the counter's usual choice, and puts it in the toaster. While the toaster runs, the worker stands beside it. Clean trays for the next orders sit at a nearby station; whether preparing them during this interval would be safe and useful has not been tested.

When the toast is ready, the worker carries the plate to a second counter for butter. Three opened milk containers from earlier preparation crowd that counter, though no current order needs milk. The worker opens one drawer, then another, looking for a knife. The toast sits on the counter while they search for a knife to spread the butter. After finding it, they butter the slice and leave it uncut. On the way back to the customer, the worker steps over an appliance cord stretched across the walking route.

The customer points out that the bread is wrong and sends the plate back. The worker must make the order again while the customer waits.

Classify the wastes the story establishes, identify one category that still needs more evidence, and address the cord hazard. For each uncertain explanation, state what observation would help distinguish it. Write your diagnosis before reading the candidate revision below.

**A candidate revision to inspect.** The preparation rack holds each ticket with the full order visible. The worker reads the complete slip, selects one slice of wholemeal bread, and leaves it uncut. The cord is secured outside the walking route before service resumes. Butter and the tools actually needed for breakfast orders are placed where they can be reached without carrying a plate to the second counter; unneeded open containers are cleared from

[2] Bruce Hamilton created *Toast Kaizen*, which inspired the use of toast making as a practice case here; see GBMP's profile of Hamilton. The breakfast-counter scene and proposed revision are written for this chapter.

the work surface according to the food-safety method. While the toaster runs, the worker prepares the next clean tray only if that task can be done safely without missing the toast or confusing orders.

Compare this sequence with your diagnosis. Which evidenced trips, searches, errors, and extra steps does it address, and which could remain? Would the new placement create another safety or quality problem? Under comparable demand, track correct orders, elapsed time from order to delivery, and unnecessary trips before deciding whether the change helped. This is a proposed method, not a reported test result.

> Reference Check: Breakfast Service
>
> The wrong bread is a defect; the avoidable carry is transportation; searching is motion; and unnecessary open supplies are inventory. The finished toast waiting while the worker searches is a delay to the order. The obscured ticket supports investigating the handoff rather than assuming the worker ignored a clear instruction. The scene does not establish overproduction, extra processing, or non-utilized talent. To investigate those categories, observe preparation quantities, the purpose of added steps, and what happens to worker suggestions. The cord requires a safety response. The worker standing beside a running toaster does not by itself establish avoidable waiting; a second task must be safe and compatible with the first.

> Waste Diagnosis Brief
>
> Choose one observed loss from breakfast service or a process you know. Record the observation, its waste classification, one supported explanation, one competing explanation, and the next evidence that would help distinguish them. Record your response in your own notes or course workbook.

### 10.4. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Understand) Define four of the Eight Wastes in your own words and give one realistic example of each from a service process.

---

**2.** (Analyze) In Wash n' Fold, mean release wait is 501.7 minutes; five Fold corrections were recorded among 39 orders; six of 21 eligible orders miss the

promise. Complete a Waste Diagnosis Brief: identify two evidenced wastes, one supported explanation, one competing explanation, and the next observation that would help distinguish them. Explain why these observations do not prove overproduction.

---

**3.** (Analyze) A team proposes buying faster equipment before mapping or measuring waste. Analyze this proposal using the least-cost-way principle.

---

**4.** (Understand) Define each of these waste pairs in one sentence each and state one practical difference:

1. Waiting vs. Inventory,
2. Motion vs. Transportation,
3. Defects vs. Overprocessing.

---

**5.** (Analyze) In a claims process, analysts repeatedly request missing fields from customers. Classify the primary waste and one likely secondary waste, then justify both choices.

---

**6.** (Analyze) A manager says, “If everyone stays busy, waste is low.” Analyze why this claim can be misleading in a queued system.

---

**7.** (Understand) Explain why waste classification must be tied to customer value and process boundary, not only local team effort.

---

**8.** (Apply) For one service process you know, list two likely “invisible” wastes that customers feel but teams often normalize.

---

**9.** (Analyze) The Wash n' Fold team could place supplies at each washer or change the serial release rule. Which observed or suspected waste would each change address, and which seems more likely to improve the 5:00 p.m. customer promise? Name the evidence you would check before choosing a test.

### 10.4.1. Journal and Reflect

Write down one waste category that you think is easiest to overlook in your own environment and one sign you would look for to make it visible.

Which empty category tempted you to invent an example? Explain how you would defend "not evidenced" to someone who expects a completed box for every waste.

Identifying waste shows where the process loses time or effort; the Waste Diagnosis Brief preserves what is observed, what is inferred, and what must be checked next. Chapter 11 uses that brief to choose and specify a small operating design: how work moves, what authorizes it to start, and what workplace conditions will support the new method.

## 10.5. Chapter Summary

- The Eight Wastes are most useful as a way of reading a real process, not as an abstract memorization exercise.
- In Wash n' Fold, release waiting and correction work are evidenced; the size of the release delay points toward a rule to investigate, not an automatic equipment purchase.
- Pair each proposed waste reduction with a local measure that tests its mechanism and a customer-outcome measure that checks whether it helped; use PCE and WIP to support that interpretation.
- The disciplined next move is not to buy speed everywhere; it is to identify the least-cost waste reductions that fit the losses the map actually shows.

# 11.

# Designing the Improved Workplace: Flow, Pull, and 5S

> "Without standards, there can be no improvement." — Taiichi Ohno
>
> "We are what we repeatedly do. Excellence, then, is not an act, but a habit." — Will Durant

## 11.1. Why This Chapter Matters

Waste diagnosis gives us a target, but it does not tell an operator what to do differently on Monday. An intervention needs an operating rule: how work moves, what authorizes it to start, where needed materials belong, and what happens when the method breaks down. This chapter develops a small, testable design using flow, pull, and 5S. Together, they turn a plausible improvement idea into work that people can actually perform and evaluate. Flow and pull address how work moves and when it begins; 5S addresses the arrangement and upkeep of the workplace.

## 11.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Lean Six Sigma Principles and Tools Use the chapter's design sequence to draft a bounded flow-and-pull rule tied to observed demand, capacity, and waste **Apply**

- Lean Six Sigma Principles and Tools Apply the 5S elements, including Safety, and document the best current method in a one-page work standard **Apply**
- Leading and Managing Projects Draft sustainment ownership, review cadence, and escalation criteria, including when 5S is insufficient to resolve a release or capacity problem **Apply**

In the previous chapter, we cataloged the Eight Wastes: the recurring patterns of non-value-added activity that drag down process performance. Identifying waste tells us *what* to target. Now we have to choose an intervention that addresses the supported mechanism without claiming more than our evidence warrants. Flow and pull define the movement and release of work. 5S supplies the physical and digital conditions that help people follow that method and notice when reality departs from it.

### 11.3. Wash n' Fold Project: Do (Continued)

We can now test whether placing detergent and softener at each Wash position reduces fetching and handoffs during setup. The shared supply jug gives us a concrete setup task to improve, but moving a jug will not abolish a 501.7-minute release wait. That distinction matters: 5S can support a better release method without being the main reason it works.

### 11.4. From Waste Diagnosis to Flow and Pull

Return to the current-state map and follow one order from Sort and tag toward Wash. The order is ready, but the release rule holds it until the active batch clears drying. That observation gives us two competing explanations for the queue: perhaps the shop lacks enough capacity, or perhaps usable capacity sits idle because the rule withholds ready work. The intervention depends on which explanation the evidence supports.

Before changing the release rule, ask what would happen if the next ready order entered Wash sooner. Would a worker, washer, dryer, folding station, safe staging area, or required piece of information prevent it from advancing? This traces the obstruction through the actual work instead of assuming that smaller batches must improve every process.

Where travel matters, sketch the physical route and consider moving sequential work closer together without compromising safety or access. Yellow Belt will introduce **spaghetti maps**, which trace actual movement on a layout so repeated trips become visible. For now, a simple route sketch and trip count

are enough to support a question about layout; the process map alone does not establish the distance traveled.

### 11.4.1. Align Capacity with Takt, Not Every Duration with Every Other Duration

Suppose a three-person service team has 480 available minutes and demand for 40 cases. Takt is $480/40 = 12$ minutes per case: the required average completion pace. One person performs each of three sequential steps, whose mean work times are 6, 14, and 8 minutes. The 14-minute step cannot sustain one case every 12 minutes under those assumptions, even if its worker stays busy all day.

First look for unnecessary work and feasible redistribution, not an instruction to hurry. If a trained colleague can take three minutes of transferable preparation from the middle step into the last step without changing the required sequence or quality, the loads become 6, 11, and 11 minutes. Each is now below Takt in this simplified comparison. They need not be identical, and measured variability, breaks, setup, and other duties can still prevent the team from achieving the intended pace.

An **operator balance chart** plots the work assigned to each operator against Takt. Bars above the Takt line identify assignments that cannot sustain the required pace under the stated assumptions.

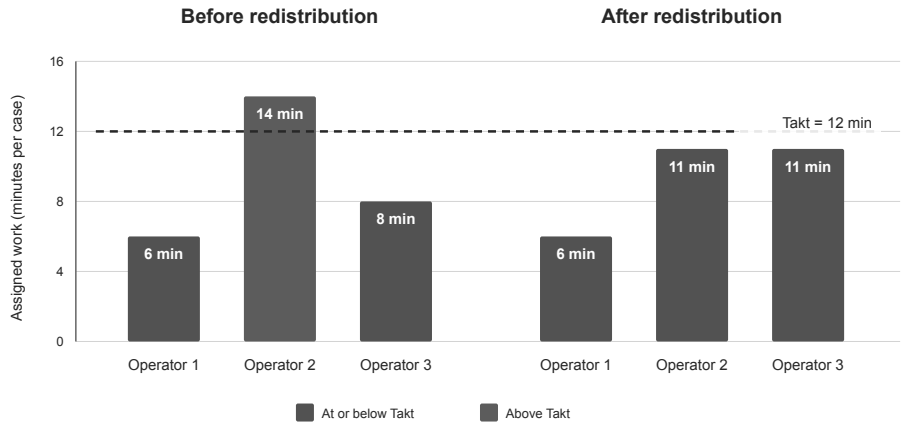



**Figure 11.1.:** Operator work before and after the proposed redistribution, compared with a 12-minute Takt.

Read Figure 11.1 from left to right. Which assignment prevents the original design from meeting Takt, and what changes after redistribution? The first chart shows Operator 2 above Takt at 14 minutes. The second shows that moving three minutes of eligible work brings Operators 2 and 3 to 11 minutes each, below Takt, while Operator 1 remains at 6 minutes.

For machines operating in parallel, compare effective output capacity with demand rather than comparing one machine's elapsed cycle directly with Takt. Two independent positions with 24-minute cycles have an ideal combined capacity of one completed unit per 12 minutes, even though each unit spends 24 minutes processing. Loading, downtime, incompatible work, or shared labor can make actual capacity lower. Do not sum automatic machine-running time and call it occupied worker time.

At Wash n' Fold, use the two-worker, six-washer, six-dryer capacity estimates from Chapter 9 and the 18.2-minute planning Takt. The first problem to investigate is the restrictive release rule, not how to make a wash cycle, a dry cycle, and folding all take 18.2 minutes. If average capacity really is insufficient, a WIP limit cannot manufacture the missing hours. In those cases, we must change the work, rebalance feasible tasks, revise staffing or equipment, or renegotiate the service commitment.

### 11.4.2. Establish Pull: Specify Permission to Start

A board with colored cards is not a pull system unless the cards govern what may begin. To see what the board must accomplish, begin with the decision an operator faces: **May I release another order?**

Figure 11.2 offers a small teaching example rather than a recommended limit for the shop. Its controlled boundary begins when an order is released to Wash and ends when the order is ready for pickup. Three orders occupy a four-order limit. The empty position therefore answers the release question: one more ready order may enter.

Which order should enter? Order A is the oldest eligible order in the visible backlog, so a first-in, first-out (**FIFO**) rule selects it unless a stated exception applies. Order F is blocked by a missing label, but the customer order has not vanished. Its card remains visible and continues to occupy one of the four positions.

What returns permission to release another order? An arrival does not. An order reaching ready for pickup leaves the controlled boundary and returns one opening. If all four positions are occupied, the next ready order waits in the visible backlog until that happens.

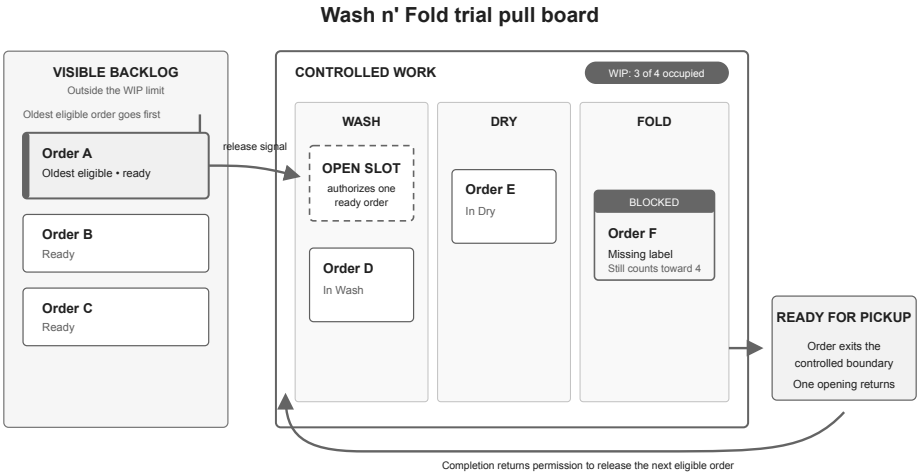



**Figure 11.2.:** A four-order pull board for Wash n' Fold.

The case has now supplied the parts of a pull rule:

1. **Controlled work:** the unit and the entry and exit events covered by the limit.
2. **Trial WIP limit:** a starting ceiling justified by observed concurrent work, available resources, safe space, necessary buffers, and completion history.
3. **Visible release signal:** the opening, card, bin, or digital signal that authorizes a stated quantity of work to enter.
4. **Selection and exceptions:** the rule for choosing the next eligible item, including justified priorities and the treatment of blocked or rework items.
5. **Signal-return event:** the specific completion that returns permission to release more work.
6. **Owner and review plan:** who maintains the rule, what the trial will measure, and what would require adjustment.

A headcount alone does not establish a WIP limit, and a hard ceiling is not the average WIP used in Little's Law. A limit that is too low can leave useful capacity idle; one that is too high can permit unnecessary queues.

Cards, empty bins, and digital boards are alternative ways of carrying a signal. The essential feature is explicit authorization tied to need and available capacity; Kanban Tool's guide to pull systems illustrates these signal types. Our later comparison will test one particular rule, not the only possible pull design.

Write a short proposal containing the controlled boundary, trial limit and its reason, release signal, signal-return event, selection rule, owner, and

follow-up measures. Keep unreleased work visible, and judge the whole customer journey rather than celebrating an empty production area with a growing backlog outside it.

### 11.4.3. Visual Controls Carry Operational Information

The pull rule has to survive contact with a busy workday. If operators must remember how many orders are authorized, search for blocked work, or ask a supervisor which order is next, the rule exists mostly on paper. A useful visual control answers an operating question at the place and moment where the decision occurs.

A card can show that one slot is available. A marked waiting lane can show which orders are authorized and which remain in backlog. A board can make the oldest ready order, the current WIP count, and a blocked item visible without a meeting. Labels at the Wash positions can show what belongs there and what needs replenishment. Each device communicates something people need to decide or act.

Color and tidiness alone do not create control. A beautiful board nobody uses to release work is decoration. A floor outline that shows where a cart belongs but gives no response when the cart is missing merely documents drift. For each visual device, ask: **What decision does this control? Who reads it? What action follows an abnormal signal?** If those answers are vague, improve the rule before adding more tape, signs, or colored cards.

### 11.4.4. Does This Change Require a Future-State Map?

Compare the proposed method with the current-state SPPM. Orders still move through Drop-off, Sort, Wash, Dry, Fold, and Stage and notify in the same broad sequence. What changes is the rule that releases an order into production, the visibility of controlled and waiting work, and the location and handling of supplies.

A second SPPM would mostly repeat the first one and could hide those operating details inside familiar boxes. For this case, keep the current-state map as the baseline and pair it with the pull board and one-page standard work. Draw a future-state map when the proposal changes the process boundary, activity sequence, handoffs, branches, or other map-level structure enough that a reader would otherwise misunderstand the proposed flow.

## 11.5. The 5S Elements

The current-state simulation records a shared detergent/softener jug in the Wash setup context: supplies are fetched, measured, and passed among positions, with setup repeated for compatible loads. Across the 39 orders, mean Wash setup is 2.5 minutes. That total does not tell us how much time each individual action takes.

Suppose we propose labeled supplies at the point of use. The proposal immediately raises practical questions. Which supplies belong at each position, and which should remain shared? Where can they sit without creating an awkward reach or obstructing access? How will operators notice a leak, a missing label, or stock approaching empty? What method should everyone follow, and who will restore the arrangement when it drifts?

The 5S elements give us a disciplined way to answer those questions. They come from the Toyota Production System and are commonly translated into English as **Sort, Set in Order, Shine, Standardize,** and **Sustain**. Many teams explicitly add **Safety**, although safety belongs in every decision from the start. The sequence is a useful starting point rather than a one-way checklist: decide what belongs before arranging it, clean and inspect the area, agree on the best current method, and build routines that keep it workable. As the work teaches the team something new, return to an earlier decision.

The decisions below develop the point-of-use supply proposal; they are not a report of changes implemented in a real shop.

| Proposed change | Local check | Customer-outcome check |
|---|---|---|
| Put needed supplies within reach | Retrieval and setup time; dosing errors and handling problems | End-to-end lead time and eligible-order promise violations |
| Keep supplies available and correctly placed | Missing supplies, replenishment interruptions, and misplaced items | Whether any local gains also improve promise performance |

### 11.5.1. Sort

Putting every available product beside every washer would eliminate some fetching and create a new stock problem. Containers would occupy space, expire, leak, and require replenishment whether that position regularly needed them or not. **Sort** asks what the work actually requires: keep needed tools, materials, and information available, and review what does not belong.

One commonly used technique to achieve this is called Red Tagging. The idea here is that we place a colored tag, sticker, or some other visual cue on all of the tools and materials in the workspace. Every time someone uses a tool or material, they remove the sticker. After a defined observation period, review items that still have a tag with the people responsible for the work. An unused item is a candidate for review, not automatic disposal: occasional tasks, emergencies, and required equipment may not have appeared during that window. Decide what must remain accessible, what can move to longer-term storage, and what can be discarded.

For homely examples, think of your kitchen cabinets. A good cabinet organization scheme should have all the materials you use daily like plates, cups, and silverware easily accessible, with less frequently used items like the Bundt cake pan or cookie cutters stored somewhere less convenient.

In a transactional environment, when you find yourself going to an internal shared hard drive or combing through a SharePoint library full of outdated materials, this is a sure sign that you're experiencing the waste of motion and would benefit from a "Sort" on your work materials.

For our supply proposal, ask the operators what each Wash position needs for its usual work and its exceptions. Determine which supplies belong at each position and which should remain shared. Duplicating everything would replace a retrieval problem with excess stock and more containers to maintain.

### 11.5.2. Set in Order

Knowing what belongs does not tell us where to put it. A detergent container can be present and still require repeated walking, obstruct a path, or force an awkward reach. **Set in Order** gives each needed item a specific location chosen around sequence, access, visibility, and safe handling.

While it is difficult to give universally applicable, general guidelines about the right way to set up your workstation, even in manufacturing, there are a couple of principles to consider that may help your thinking.

1. Try to make work areas "cellular," meaning that each workstation in the work area produces only one product (or product family), so that you can gather all the tools needed to make that product in one place with a minimum of transportation of materials back and forth.
2. Try to keep the size of the workstation small enough to encourage continuous flow rather than the production of large batches.

3. Try to position all materials and tools ergonomically and within easy reach of the operators, to minimize the time they have to spend stretching or reaching.
4. Consider adopting a straight-line or U-shaped layout for work areas. If the final product needs to end up back at the same physical location where the order arose, a U-shaped layout is a natural solution.
5. Try to make it easy for someone to determine just by looking at the work layout where each product is in the production process.

If you follow these general guidelines, you'll probably hit on an acceptable ordering of the workspace pretty quickly. Through experience and experimentation, this can be refined with the assistance of the operators.

If we're working on a project in a transactional process, such as processing an invoice or generating a report, then applying these principles relates more to the motion involved in navigating between different screens in a piece of software. Setting a virtual workplace in order follows most of the same principles as setting a factory floor in order: again, we're trying to gather like pieces of work together and make it as easy as possible for people to find the information they need quickly.

For the supplies we decide to place at each position, choose an accessible location with the operators and make the contents and intended location clear. Check that reaching for them does not obstruct access or create awkward handling. Try the arrangement through the actual setup motions before treating the layout as settled.

> Pro Tip: Design With the People Who Do the Work
>
> Operators know where the written process collides with a heavy bag, a rush of arrivals, an unreliable label, or a task that quietly requires two hands. Involve them while observing the current method, choosing the intervention, trying the arrangement, and writing the standard. This is not merely a way to obtain better facts, although it does that. People are more likely to support and continue a method they helped test and improve than one delivered after the project team has already made the important decisions. Participation does not mean accepting every preference or avoiding difficult changes. It means treating operators as sources of operational knowledge and as partners who will have to make the method work after the project closes.

### 11.5.3. Shine

The proposed containers now have sensible locations, but a slow leak or damaged label could remain hidden by residue and clutter. **Shine** means cleaning and inspecting the work area so abnormalities become visible. In manufacturing, this means cleaning machines, floors, work surfaces, and tools so that leaks, wear, misalignment, and safety hazards become immediately visible. In service and transactional work, "shine" means maintaining the digital workspace too: reviewing stale tickets, archiving obsolete files, clearing broken links, and keeping the active queue transparent. Resolve or escalate outstanding work; close a ticket only when the work is complete or confirmed no longer required. Age alone is not a reason to make a customer request disappear.

The point is not cosmetic perfection; the point is to make the process visible. A clean workspace makes abnormalities easier to detect and respond to. For the supply arrangement, include inspection of containers and their locations so spills, leaks, or deterioration can be noticed and addressed. These are conditions to inspect for, not incidents established by the simulation.

### 11.5.4. Standardize

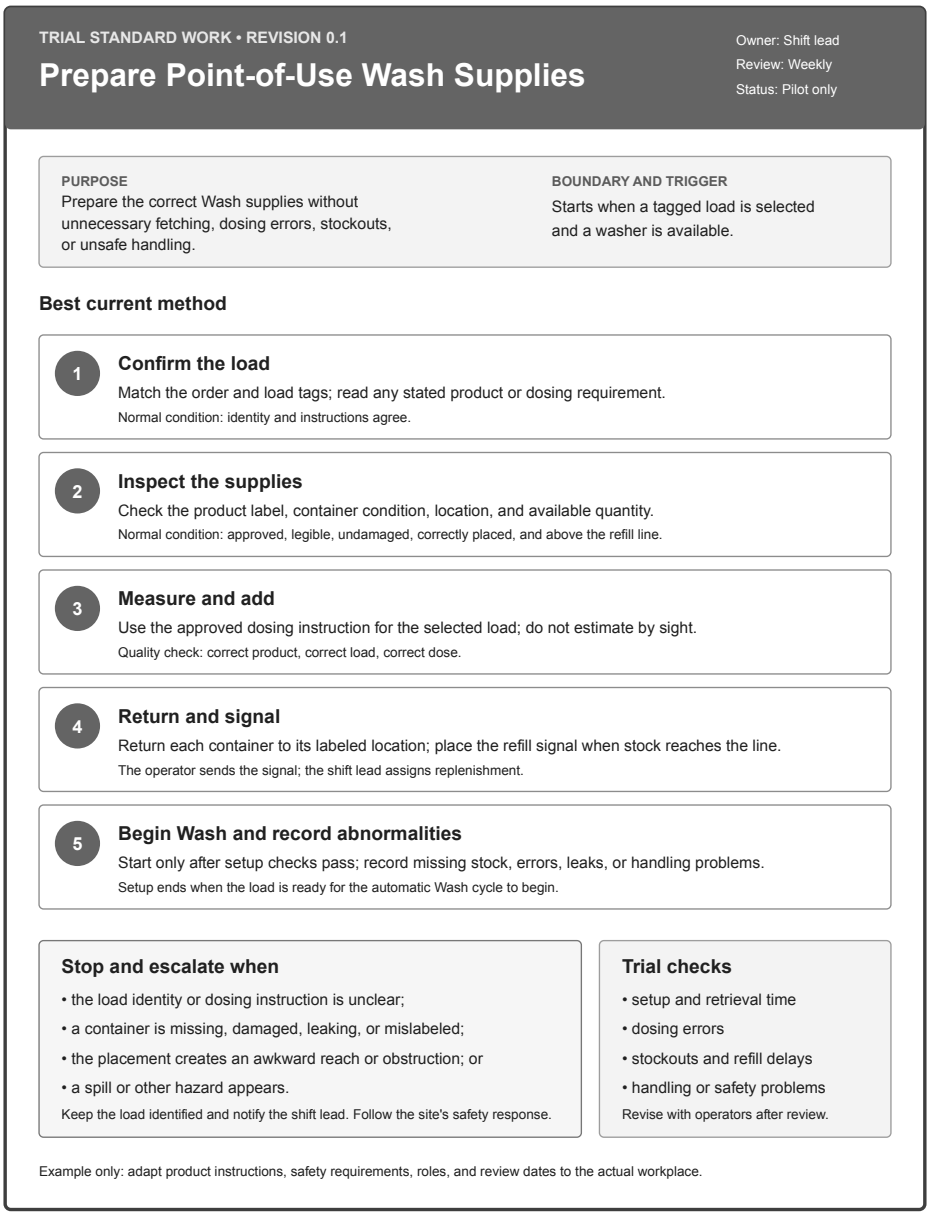
Suppose one operator returns the detergent after each load, another leaves it beside the washer, and neither knows who should refill it. The new location exists, but the method still depends on memory and private interpretation. Labels alone have not standardized the work.

**Standardize** records the best current method clearly enough that two trained operators would perform the important parts in the same way. The standard names the trigger, sequence, responsible role, expected condition, quality and safety checks, abnormalities, and escalation route. It creates a stable baseline from which the team can detect drift and learn; it does not make the first answer permanent.

A standard is the team's best current method, not a commandment delivered from outside the work. The people who perform the process know where instructions become awkward, where exceptions arise, and which shortcuts signal that the written method does not fit reality. Build the first version with them, make it specific enough to use, and revise it when evidence reveals a better way.

Figure 11.3 shows one way to fit the Wash supply method on a single page at the place of work. Read it from the purpose and trigger into the five steps, then inspect the stop conditions and trial checks. Notice what the page does

not do: it does not bury the operator in background theory or pretend to replace product instructions and site safety procedures.



TRIAL STANDARD WORK • REVISION 0.1

## Prepare Point-of-Use Wash Supplies

Owner: Shift lead
Review: Weekly
Status: Pilot only

**PURPOSE**
Prepare the correct Wash supplies without unnecessary fetching, dosing errors, stockouts, or unsafe handling.

**BOUNDARY AND TRIGGER**
Starts when a tagged load is selected and a washer is available.

**Best current method**

1. **Confirm the load**
   Match the order and load tags; read any stated product or dosing requirement.
   Normal condition: identity and instructions agree.
2. **Inspect the supplies**
   Check the product label, container condition, location, and available quantity.
   Normal condition: approved, legible, undamaged, correctly placed, and above the refill line.
3. **Measure and add**
   Use the approved dosing instruction for the selected load; do not estimate by sight.
   Quality check: correct product, correct load, correct dose.
4. **Return and signal**
   Return each container to its labeled location; place the refill signal when stock reaches the line.
   The operator sends the signal; the shift lead assigns replenishment.
5. **Begin Wash and record abnormalities**
   Start only after setup checks pass; record missing stock, errors, leaks, or handling problems.
   Setup ends when the load is ready for the automatic Wash cycle to begin.

**Stop and escalate when**

- the load identity or dosing instruction is unclear;
- a container is missing, damaged, leaking, or mislabeled;
- the placement creates an awkward reach or obstruction; or
- a spill or other hazard appears.

Keep the load identified and notify the shift lead. Follow the site's safety response.

**Trial checks**

- setup and retrieval time
- dosing errors
- stockouts and refill delays
- handling or safety problems

Revise with operators after review.

Example only: adapt product instructions, safety requirements, roles, and review dates to the actual workplace.

**Figure 11.3.:** Completed one-page trial standard work for point-of-use Wash supplies.

Download the completed Wash setup standard (US Letter PDF) to inspect beside your own draft.

Use Figure 11.4 when your proposal creates or changes a repeatable operating method that another person needs to perform consistently. Keep

the standard to one page. Do not create one merely to complete a form: a location change or one-time removal may need an owner and check without needing procedural standard work. If a repeatable method cannot be stated that plainly, the team may not yet agree on it or may be trying to force several distinct tasks into one document.

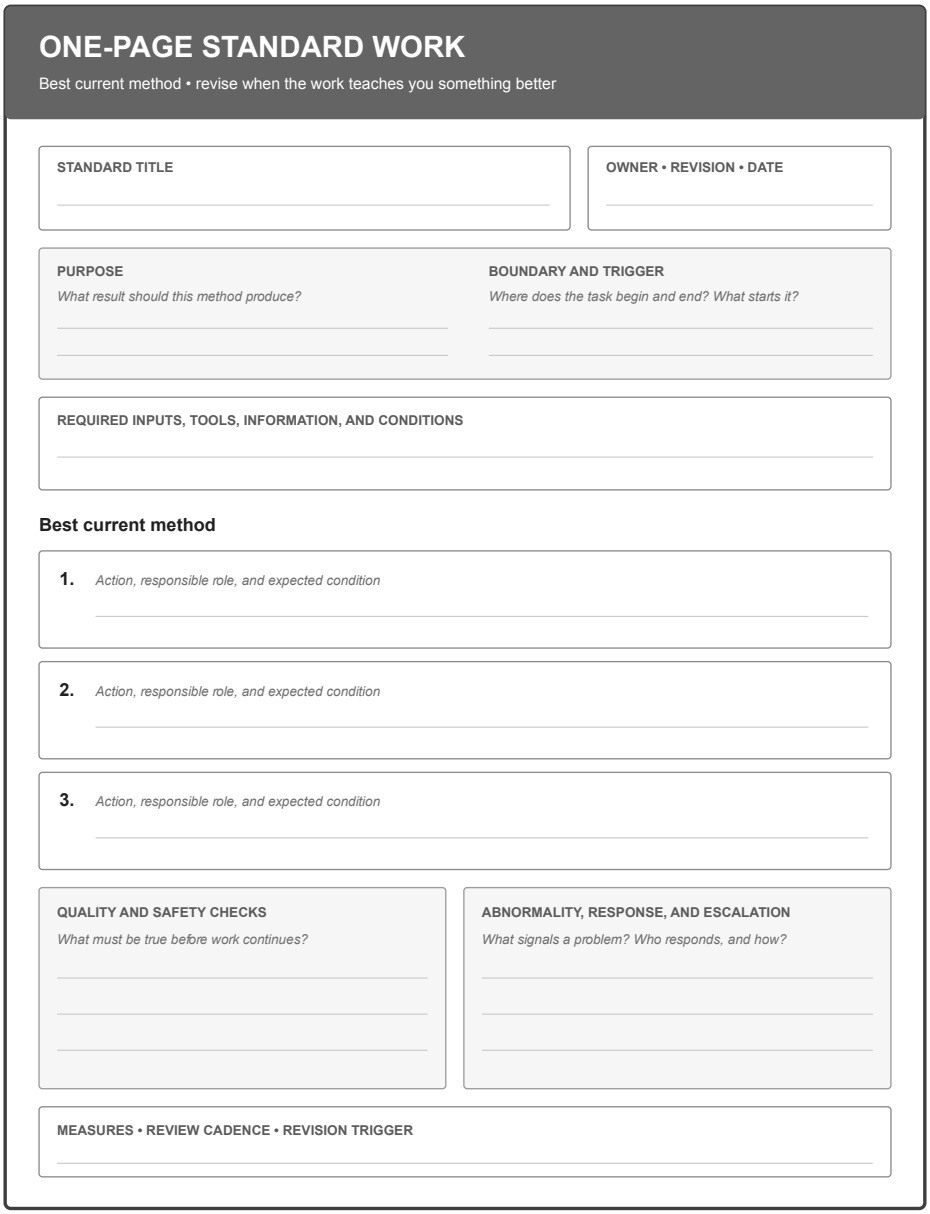



**Figure 11.4.:** Blank one-page standard work template.

Download the blank one-page standard work template (US Letter PDF) when your proposal needs a repeatable method. Print one copy per method and

fill it with your own process's conditions; do not copy the Wash setup instructions into an unrelated process.

The pull rule needs the same precision even though its visual control is a board rather than a supply label. An open position triggers release of the oldest eligible order; a blocked order remains visible and keeps its position; persistent blocking goes to the shift owner. That detail keeps two well-intentioned operators from inventing different systems halfway through the trial.

### 11.5.5. Sustain

Suppose the arrangement works during the project week, then the refill cards disappear and containers begin migrating between washers. The design did not become self-maintaining when the labels were attached. **Sustain** creates the ownership, review, correction, and revision routines that keep the method alive after the improvement event.

To sustain gains, use simple management routines:

- A short recurring audit (daily or weekly) with a small checklist.
- A named owner for each area/process.
- Fast correction of drift when standards are not being followed.
- Periodic review to update standards when the process changes.

In other words, sustain is about behavior, not just layout. The organization has to reward disciplined adherence to standards and continuous small improvements over one-time cleanup events.

A trial sustainment routine could assign one person each shift to check availability, labels, and placement, with a weekly review of setup times and interruptions. Record missing or misplaced supplies and replenish or correct them through the agreed method. Repeated shortages should trigger a review of the replenishment rule; repeated misplacement may mean the chosen location does not fit the work. The owner needs a way to get those problems resolved, not just a checklist to initial.

That last distinction protects sustainment from becoming audit theater. A high checklist score is evidence that recorded conditions met the checklist; it is not evidence that the process improved. If setup remains slow while every box is checked, the checklist may be measuring the wrong things. If the same exception appears every week, the cause may be an unrealistic standard, insufficient replenishment capacity, or a problem the named owner lacks authority to correct. Treat recurring noncompliance as information about the design as well as behavior. Review performance beside compliance, give

the owner a clear escalation path, and change the standard when the work teaches you that it should change.

> Pro Tip: Leadership Has Work to Do Too
>
> A leader cannot announce 5S, inspect a score, and leave operators to supply the time, materials, authority, and persistence the method requires. Leaders need to visit the work, listen to the people doing it, remove obstacles they cannot remove themselves, and visibly follow the same standards they expect others to follow.
> Toyota describes its operating philosophy as resting on **Continuous Improvement** and **Respect for People**, with success created through individual effort and teamwork (Toyota Way 2001). The Lean Enterprise Institute makes the practical consequence plain: visual management and standard work fade when leaders treat respect and engagement as optional and disappear once the implementation team leaves ("Leading with Respect Is Not a Soft Skill"). Visible participation matters because it shows that maintaining and improving the workplace is part of the work, not cleanup delegated to people with less authority.

#### 11.5.6. Safety

Many teams explicitly include Safety as a sixth S because unsafe work is always defective work. An injury, near miss, ergonomic strain, or unmanaged hazard is both a human cost and a process failure.

In practical terms, safety should be integrated into each of the other S's:

- **Sort:** remove hazardous clutter and obsolete materials.
- **Set in Order:** place tools/materials to reduce overreaching, twisting, and unnecessary lifting.
- **Shine:** use cleaning/inspection to detect leaks, wear, and hazards early.
- **Standardize:** include safety-critical steps and checks in standard work.
- **Sustain:** audit and correct safety drift continuously.

For the point-of-use proposal, these checks accompany the choices about supplies, locations, inspection, and replenishment. Address a handling problem when it is identified rather than waiting to reach a sixth step.

#### 11.5.7. Plan the Supply Trial

The proposal is now specific enough to test: chosen supplies have labeled locations, a preparation and replenishment method, and someone respon-

sible for reviewing how it works. Treat the trial as a small experiment in the everyday sense: state what will change and decide before seeing the result what evidence would support retaining, revising, or abandoning the arrangement.

First define the local measure. For this trial, setup begins when an operator starts retrieving and preparing Wash supplies and ends when the load is ready for the automatic Wash cycle to begin. Use the same boundary before and after. Record enough setups across ordinary work to see whether a single unusually easy or difficult load is driving the result, and note relevant differences such as unusual dosing requirements or missing stock. The team might choose one representative week before and one week during the trial, with a minimum number of completed setups in each period. The exact number is a planning choice based on available work; it should be stated before the team sees the answer it prefers.

The trial also needs guardrails. Use approved containers and dosing instructions, keep required product information visible, do not place supplies where they create awkward reaches or spill hazards, and stop or revise the arrangement if errors, leaks, or handling problems increase. Speed does not excuse a quality or safety failure.

Before starting, write the decision rules in plain language. Retain the arrangement if setup and retrieval improve without worse dosing, replenishment, or safety results. Revise it if the location helps but shortages, labeling, or handling create a correctable problem. Abandon it if the arrangement produces no useful local change or creates a risk that cannot be acceptably controlled. Those rules are hypotheses and guardrails rather than universal numerical thresholds, but writing them early makes the later judgment more honest.

Record the local measure, customer-outcome measure, guardrails, and decision rules with the proposed method. Chapter 12 will compare the observations and decide what the evidence supports.

#### 11.5.8. Assemble the Trial Proposal

The pieces belong in one proposal rather than in separate worksheets that the team must mentally reassemble. The Wash n' Fold proposal below states the primary change, its supporting changes, the evidence each component requires, and the rules for acting on the result.

| Proposal field | Wash n' Fold trial |
|---|---|
| Problem and supported mechanism | A 501.7-minute mean release wait dominates the baseline; the serial release rule may hold ready orders while usable capacity is available. |
| Primary operating change | Replace serial batches with FIFO pull across the released-production boundary, beginning with a trial limit of 12 customer orders. Completion returns one opening; blocked work remains visible and occupies its position. |
| Supporting changes | Request notification when production completes; place approved Wash supplies at labeled points of use with a defined replenishment method. |
| Owner and trial period | Shift owner maintains the release signal; a named Wash-area owner maintains supplies; test across comparable Fridays before deciding. |
| Local mechanism measures | Release wait and idle positions while ready orders wait; notification wait; Wash setup and retrieval time. |
| Customer-outcome measures | Eligible-order promise violations and P90 operating lead time under the Chapter 9 boundary and cohort rules. |
| Guardrails | All open customer orders remain visible; correction does not worsen; blocked work is not hidden; dosing, replenishment, handling, and safety remain acceptable. |
| Decision rules | Retain a component when its mechanism improves without violating guardrails; revise a correctable rule or condition; abandon a component that adds no useful change or creates unacceptable risk; defer judgment when its required evidence is missing. |
| Standard work | Required for the repeatable release, notification, and supply methods; unnecessary for a one-time change that creates no continuing operating method. |

This proposal is the input to Chapter 12. The after-state evidence should answer each row separately where possible; a favorable package result cannot substitute for a missing component measure.

#### 11.5.9. Another Application: Quality Improvement (Transactional)

Consider a hypothetical claims team whose submissions are returned because attachment naming is inconsistent. The work itself was technically correct, but routing and review failed when files were hard to identify.

The proposed actions are:

- **Sort:** archive obsolete templates and remove duplicate forms.
- **Shine:** clean broken links and stale folders weekly.
- **Standardize:** enforce one file naming convention and one intake checklist.

Expected effect: lower rework loop rate and fewer approval delays.

A plausible first result would be fewer attachment-related returns; end-to-end lead time might not change if another queue still dominates. That is a prediction to test, not an observed before/after result. Count returns and record their reasons, then compare lead times under the same boundary and comparable demand.

### 11.6. When 5S Is Not Enough

5S is powerful, but it is not a cure-all. If a process is structurally undercapacity, 5S may reduce searches and avoidable interruptions, but it does not supply the missing capacity.

For example, if dryers are fully loaded all day and demand exceeds available machine hours, better organization helps, but queue inflation will remain. In that case, investigate the capacity or scheduling problem while reducing avoidable workplace friction where it helps.

Use this rule of thumb:

- If search/retrieval and interruptions are evidenced, target their causes and plan how to sustain the change.
- If a release rule dominates the wait, investigate that rule while preparing the workplace to support the revised flow.
- If demand genuinely exceeds available capacity, do not promise that better organization alone will close the gap.

At Wash n' Fold, visual waiting locations, clear release signals, and an auditable work limit could support a different release method. They do not choose that method for us. Nor should we postpone investigating the dominant delay until every drawer is perfectly arranged.

Choose 5S actions where workplace conditions contribute to the observed problem, and combine them with other changes when the evidence calls for it.

**Reflection Question: Apply 5S to Wash n' Fold**

Return to your Wash n' Fold current-state map and identify one practical 5S action for each category (Sort, Set in Order, Shine, Standardize, Sustain, and Safety).
For each action, briefly note:

1. which waste(s) it is intended to reduce,
2. one local measure that would test the change, and
3. one customer-outcome measure that would check its benefit.

Label actions that need further observation before you could justify trying them. Show how safety is addressed throughout the proposal. Record your response in your own notes or course workbook.

**Answer Key: 5S on Wash n' Fold (Reference)**

Example countermeasures (one possible set):

- **Sort:** review duplicate or obsolete tags and supplies with operators before removing them from the intake counter.
- **Set in Order:** create fixed labeled locations for tags, bags, labels, and staging bins.
- **Shine:** end-of-shift clean-and-inspect routine for intake, folding table, and transfer area.
- **Standardize:** one standard intake script, tag format, and handoff checklist.
- **Sustain:** daily 5-minute audit with owner checklist and weekly review.
- **Safety:** mark transfer path and wet-risk zones; keep loads within safe carry limits.

For point-of-use supplies, check setup time first. For status visibility, check notification delay; for a release standard, check compliance and work counts. For these actions, use end-to-end lead time or eligible-order promise violations to check customer benefit; PCE and work counts can help explain the result. There is no guaranteed order in which those system measures will improve, and the model does not establish that every suggested action is needed.

### 11.7. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** Understand In your own words, explain the difference between **Sort** and **Set in Order**. Give one practical example of each from either a physical or digital workspace.

---

**2.** Apply Complete a Wash n' Fold Trial Proposal using the fields modeled above. Identify the primary operating change, any supporting flow or 5S changes, mechanism, owner, local and customer measures, guardrails, and retain/revise/abandon criteria. State which parts require standard work and why.

---

**3.** Apply A team reports that after a cleanup day, the area looked great for one week, then gradually returned to clutter. Which parts of 5S are most likely weak in their approach, and what evidence supports your diagnosis?

---

**4.** Evaluate Evaluate this statement: "5S is mostly about neatness, not process performance." Do you agree or disagree? Defend your answer with one operations argument and one customer-impact argument.

---

**5.** Understand Define each 5S element in one sentence and name one common implementation mistake for each.

---

**6.** Apply Pick one workstation and draft a 5-day 5S trial plan with clear ownership for each day. Address safety throughout and include a point at which observations could lead you to revisit an earlier decision.

---

**7.** (Apply) Write a simple sustain audit checklist with five yes/no checks and one weekly review question. Name its owner and an escalation trigger for a high release queue that persists despite compliance with the workplace standard.

---

**8.** (Understand) Explain why Safety is treated as integrated with 5S in this chapter instead of a separate afterthought.

---

**9.** (Apply) If your proposal creates or changes a repeatable operating method, complete the one-page standard work template for that method. Include its purpose, boundary, trigger, owner, sequence, quality and safety checks, abnormal conditions, escalation route, measures, and review cadence. If it does not require standard work, explain why an owner and review check are sufficient.

---

**10.** (Apply) Select a digital workspace (shared drive, ticket board, or repo). Apply 5S language to it and list one action for each S.

---

**11.** (Apply) A claims team has three dedicated operators, 480 available minutes, and demand for 40 cases. Its sequential work times are 6, 14, and 8 minutes per case. Calculate Takt and explain the capacity mismatch. Describe one conditional rebalancing option, then write a trial pull rule with a limit, release signal, signal-return event, and treatment of blocked work. Name one measure that would warn you the limit was too low.

---

**12.** (Apply) State what you will measure to check your flow, pull, and 5S trial, and what result would make you retain, revise, or abandon the proposed change.

### 11.7.1. Journal and Reflect

Write about one place where clutter, searching, or inconsistency has become so normal that people no longer notice it. What would you want to observe before deciding whether 5S is the right response?

Suppose a clean, orderly shop still misses its customer promise. What evidence would make you stop prescribing more cleanup and investigate the release rule instead?

With waste identified and an intervention designed, the next step is to close the PDCA loop. In the next chapter, we run the Check and Act phases on the Wash n' Fold project before the capstone asks you to apply the full cycle to a process of your own choosing.

## 11.8. Chapter Summary

- Flow removes evidenced obstructions to the next necessary step; pull states what authorizes new work to begin and keeps unreleased work visible.
- Visual controls must carry information that governs a decision or response; color and neatness alone do not create control.
- 5S is a practical method for changing the conditions that allow waste to recur, not a cosmetic cleanup ritual.
- Sort, Set in Order, Shine, Standardize, and Sustain provide a useful starting sequence that teams revisit as they learn; safety belongs throughout.
- Standard work records a repeatable operating method, including its trigger, owner, abnormal condition, and escalation path; use it when the change needs consistent execution, not merely to complete a form.
- Plan a 5S trial with a consistent local measure, customer-outcome measure, and stated guardrails; Chapter 12 uses those commitments to judge the result.
- Investigate capacity, release, or scheduling problems when the evidence points to them; that work need not wait for a completed 5S effort.

# 12.

# Check and Act: What Did We Actually Improve?

> "Without data, you're just another person with an opinion." — W. Edwards Deming

## 12.1. Why This Chapter Matters

A countermeasure can be sensible, well designed, and faithfully carried out—and still fail to solve the customer problem. Check is where we compare that hope with what happened. Act is where we decide what the evidence earns: retain the method, revise it, abandon it, or begin another cycle around what remains. This chapter closes the guided Wash n' Fold cycle before the capstone asks you to run one yourself.

## 12.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Data Understanding Compare before-and-after process measures using consistent boundaries, cohorts, units, and calculation methods **Analyze**
- Lean Six Sigma Principles and Tools Distinguish evidence from one matched observation, repeated descriptive evidence, and claims that require later statistical analysis **Evaluate**
- Leading and Managing Projects Make an Act recommendation that names what to retain, revise, monitor, and investigate in the next PDCA cycle **Evaluate**

- Business Understanding Present a bounded recommendation to a process owner using customer value, operating mechanism, business trade-offs, evidence limits, and a clear decision request **Apply**

Chapters 10 and 11 took us through Do: diagnose the observed waste, design a release rule, and create workplace conditions that make the proposed method executable. Now we will compare the proposed mechanism with after-state evidence. The aim is not to reward a favorable number. It is to decide what the evidence allows us to say.

## 12.3. Wash n' Fold Project: Check

### 12.3.1. Return to the Trial Proposal Before Seeing the Results

Chapter 9 established the customer problem: six of 21 eligible orders missed 5:00 p.m. on the representative simulated Friday. The biggest measured delay was the release queue before Wash. Chapter 10 separated that evidence from attractive guesses; Chapter 11 turned the diagnosis into a flow-and-pull proposal supported by 5S and standard work.

Return to the Trial Proposal from Chapter 11 before reading the tested method below. Its mechanism, risks, measures, guardrails, and decision rules were written before the result was known. That order matters: Check evaluates a prior claim rather than inventing a favorable standard after seeing the answer.

A warning is warranted: 5S alone probably will not cut it. Reducing a few setup minutes can help, but it does not remove the rule holding the next batch until the active batch has cleared drying. A good proposal addresses that delay without simply flooding downstream work or hiding unprocessed orders outside the count.

### 12.3.2. The Method Tested in the Simulation

Compare the proposal with this tested package:

- Replace serial batch release with FIFO pull: release the next sorted order when a production-work slot is available.
- Limit released production WIP to 12 customer orders, returning a slot after production is complete.
- Request staging and notification as soon as an order is production-complete rather than waiting to assemble a notification batch.

- Put labeled supplies at the point of use, reducing Wash setup while retaining a small residual setup time.

This is a staged set of related changes, not the model for a capstone that tests one bounded countermeasure. Release control is the primary intervention; prompt notification and point-of-use supplies are supporting changes whose incremental evidence must be considered separately. The work limit is a modeled trial setting, not a universal recommendation. Orders waiting for a production slot still count as open customer orders. "Immediate notification" removes intentional batching; it does not promise instant service when both workers are occupied.

No workers, machine positions, operating hours, or nominal machine-cycle speeds were added. The matched scenarios use the same arrivals, order attributes, underlying service requirements, and correction decisions. This makes the comparison useful for examining the modeled rules, although it cannot establish what an actual laundry would achieve.

### 12.3.3. A Small After-State Evidence Packet

Begin with the same representative Friday as before: run 91, selected from the current state without inspecting its improvement. The after state is the `future_pull_5s` scenario for the same seed. There are still 39 orders, of which the same 21 qualify for the promise.

Chapter 9 reconstructed one complete order by hand so the software's aggregation would be intelligible. Here, software performs that repetitive work for the after-state records. The learner's job is to verify that the output uses the intended cohort, boundary, units, and denominator, then interpret the comparison.

The boundary is still drop-off to ready-and-notified. Minute zero is 7:00 a.m.; eligibility ends at minute 120; the promise deadline is minute 600. Closed hours are omitted from the cumulative operating clock.

| Order suffix | Arrival | Before operating L/T | After ready | After operating L/T |
|---|---|---|---|---|
| 001 | 9.1 | 165.2 | 143.7 | 134.6 |
| 002 | 10.9 | 162.8 | 152.8 | 141.9 |
| 003 | 12.1 | 166.3 | 172.1 | 160.0 |
| 004 | 31.8 | 254.5 | 225.6 | 193.8 |
| 005 | 33.7 | 253.2 | 211.8 | 178.1 |
| 006 | 44.1 | 246.4 | 221.8 | 177.7 |
| 007 | 47.8 | 343.1 | 229.9 | 182.1 |
| 008 | 52.8 | 339.1 | 291.4 | 238.6 |

| Order suffix | Arrival | Before operating L/T | After ready | After operating L/T |
|---|---|---|---|---|
| 009 | 60.3 | 334.3 | 292.3 | 232.0 |
| 010 | 61.2 | 424.5 | 334.8 | 273.6 |
| 011 | 62.1 | 423.2 | 295.4 | 233.3 |
| 012 | 63.6 | 425.2 | 330.3 | 266.7 |
| 013 | 75.3 | 518.1 | 362.8 | 287.5 |
| 014 | 78.2 | 511.5 | 366.9 | 288.7 |
| 015 | 81.0 | 509.5 | 340.5 | 259.5 |
| 016 | 88.0 | 598.3 | 372.8 | 284.8 |
| 017 | 89.7 | 596.0 | 386.3 | 296.6 |
| 018 | 97.4 | 591.4 | 412.8 | 315.4 |
| 019 | 100.0 | 682.0 | 408.8 | 308.8 |
| 020 | 103.1 | 678.4 | 426.0 | 322.9 |
| 021 | 110.5 | 676.2 | 450.5 | 340.0 |

The software-produced comparison for the 21 promise-eligible orders is:

| Eligible-order measure | Before | After |
|---|---|---|
| Mean operating lead time (min) | 423.8 | 243.6 |
| Median operating lead time (min) | 424.5 | 259.5 |
| Nearest-rank P90 operating lead time (min) | 676.2 | 315.4 |
| Orders ready after the 5:00 p.m. promise | 6 of 21 (28.6%) | 0 of 21 (0%) |

**! Audit the Output, Then Make a Provisional Judgment**

Complete Exercise 5, then write the provisional Check/Act recommendation in Exercise 6 before opening its answer. Confirm that the software used the same 21-order cohort and deadline as the baseline, and verify the after-state violation count directly from the ready timestamps. Use the supplied results for the remaining outcomes. The aim is to audit and interpret the calculation, not to repeat arithmetic already learned in Chapter 9.

| Supplied measure, all 39 orders | Before | After |
|---|---|---|
| Mean operating lead time (min) | 662.6 | 314.3 |
| PCE, total VA / total operating L/T | 13.9% | 29.2% |

| Supplied measure, all 39 orders | Before | After |
|---|---|---|
| Correction occurrences | 5 | 5 |

These figures are derived from the displayed one-decimal occurrence timestamps. After-state VA time sums to 3,582.6 minutes and operating L/T to 12,256.6 minutes. The baseline VA sum was 3,582.3; that tiny difference is timestamp-rounding arithmetic, not a claim of changed service requirements. Keep this all-order readout separate from the 21-order promise cohort.

### 12.3.4. Check the Local Change and the Customer Result

The package changes both the release method and the supply arrangement, so examine it at two levels. For the supply trial, compare retrieval and setup under the current and proposed methods using the boundary defined in Chapter 11. Check dosing errors, replenishment interruptions, and handling or safety problems beside the timing. A faster setup that creates a new problem is not an improvement.

Then return to end-to-end lead time and eligible-order promise violations. Most of the baseline delay occurred while the release rule held the next batch, not while somebody measured detergent. A real setup saving could therefore leave the customer promise largely unchanged, while a release-rule change could improve the customer result even if setup barely moved.

The simulated packet reports the combined result of several changes and does not supply a separate observed setup-time series for the point-of-use trial. Do not use the package result to pretend that the supply arrangement alone caused the lead-time change. The proper component decision here is **defer judgment and collect the missing setup evidence**. In an actual pilot, retain the arrangement if the local measure improves without worse dosing, replenishment, or safety results; revise it when a correctable problem appears; abandon it when it produces no useful local change or creates a risk that cannot be acceptably controlled. Those judgments apply the decision rules written during Do.

### 12.3.5. What This Packet Does Not Establish

A better result on one matched Friday is evidence about that simulated Friday. It does not guarantee zero late orders next week, establish a real-world effect, or separate the contributions of four changes made together. The broader evidence in the answer to Exercise 6 follows the same 52 underlying

Friday demand streams through each scenario. That gives us one simulated calendar year rather than a lucky afternoon, but it still comes from a model.

> **A Question We Are Not Yet Equipped to Answer**
>
> Repeated observations give us a better view of ordinary variation than one Friday can. They still do not tell us, by inspection alone, how much uncertainty surrounds an estimated improvement or how confidently an observed difference can be separated from ordinary variation. Later statistical methods will help us ask those questions rigorously.

> **When the Result Is Not What You Expected**
>
> Sometimes an intervention does not produce the result you expected. You may have chosen the wrong intervention. The intervention may have addressed one real problem while another constraint, defect, or source of delay remained beneath the surface. The process may already have been close to the best performance it could achieve under its current conditions, leaving little room for the proposed change to help. The method may also have been carried out inconsistently, or ordinary variation may have obscured a real but modest effect.
> At White Belt, you do not yet have all the tools needed to decide confidently which explanation is most likely. You can still do disciplined work: verify what was actually implemented, compare consistent measures, report the unexpected result honestly, and identify the next observation or test that would separate the competing explanations. Later belts add deeper diagnostic, experimental, and statistical methods for making that judgment. Until then, do not force an appealing story onto an uncooperative result. Carry the uncertainty into the next PDCA cycle.

## 12.4. Wash n' Fold Project: Act

### 12.4.1. Standardize What the Evidence Supports

An Act recommendation should say what you would retain, what you would monitor, and what would make you reconsider. In a simulation, "retain" means carry the method into the next modeled test or propose a bounded real pilot. Do not describe the result as an implemented gain in an actual business.

Assign an owner for the chosen release standard and supply locations. Review open customer orders as well as released production WIP, and track promise failures alongside setup, correction, and notification delays. A low count inside the production-work limit is no success if customers are merely waiting outside it.

State an escalation trigger before the next test: for example, the eligible-order violation rate exceeding the 10% goal over a defined review window, or repeated release-rule noncompliance. The window and denominator must be explicit; one small day with no failures cannot establish permanent reliability.

Compare this recommendation with the proposal you wrote before seeing the tested package. What did the evidence change? A well-reasoned alternative remains worth testing even if it differs from this worked example.

### 12.4.2. Presenting the Recommendation to the Owner

The following worked dialogue models a presentation to an owner. It shows how to request a trial, answer questions about the evidence, and incorporate a worker's objection.

The project team does not open with the simulation model or the PCE calculation. It opens with the decision the owner has to make.

"We recommend a controlled trial of a pull release rule, prompt notification, and point-of-use supplies," the presenter says. "The aim is to reduce missed 5:00 p.m. promises without increasing corrections or exhausting the staff. We are asking you to authorize a small real pilot, not to accept the modeled result as a guaranteed savings."

The owner points at the row showing zero late eligible orders on Friday 91. "Then why not say the problem is solved?"

"Because that was one simulated Friday. Across 52 modeled Fridays, the complete package still produced five late orders. The result is encouraging, and the matched comparisons support the mechanism, but no customer order has moved through your shop under this rule yet."

The owner asks which part of the package did the work. The team shows the incremental scenario comparison. Release control produced the largest modeled reduction in operating lead time; prompt notification and point-of-use supplies added smaller gains. The team also states what the simulation did not improve: correction decisions were held fixed, so the package supplies no evidence that the tagging and relabeling problem has been solved.

"Could we get the same result by buying another washer?" the owner asks.

"Possibly, but the current evidence does not identify washer capacity as the dominant cause of the missed promise. The largest observed delay came from the rule that held work before Wash. Buying capacity before testing the release rule would cost more and might leave that delay intact."

One of the workers then raises a practical objection. Twelve released orders may be manageable on an ordinary Friday, but a few unusually large orders could crowd the folding stations and make the limit unsafe or misleading. The team adds order size and blocked-work visibility to the trial checks and agrees that the workers can stop the release when the defined conditions are not met. The modeled limit of 12 becomes a starting hypothesis to examine, not a command imported from a spreadsheet.

The owner authorizes planning for a bounded pilot, with the final release limit set only after the workers review recent order sizes and available space. One person will own the release signal and another will verify the ready-and-notified timestamps. The review will compare late-promise rate, operating lead time, all open orders, correction occurrences, and worker-reported overload using the same definitions as the baseline. If quality worsens, work becomes unsafe, or the release rule repeatedly breaks down, the team will stop and revise the trial.

A compact owner brief should state:

- **Decision requested:** what authority, resources, or trial approval is needed now.
- **Customer and business problem:** who is affected and why the result matters to the firm.
- **Proposed mechanism:** what will change in the work and why it could affect the problem.
- **Expected benefit and burden:** which outcomes may improve, what effort or cost the change requires, and who carries it.
- **Evidence and limits:** what was observed, calculated, or simulated and what remains unknown.
- **Measures and review rule:** what will be monitored, who owns it, when the result will be reviewed, and what would stop or revise the trial.

The owner's questions help validate the recommendation. A recommendation becomes more credible when it survives questions about cost, feasibility, worker burden, and the difference between a modeled result and an implemented one.

### 12.5. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Evaluate) Suppose your pilot shows improved lead time but a slight rise in defect/rework rate. Would you standardize now, revise and re-test, or roll back? Defend your decision.

---

**2.** (Apply) Write a one-paragraph Act-phase standardization note for one improvement that worked in your case. Include owner, trigger, review cadence, and the condition that would reopen the method for revision.

---

**3.** (Apply) Draft the opening Plan statement for the next PDCA cycle based on what remained unresolved. Include one problem statement sentence and one SMART goal sentence.

---

**4.** (Evaluate) Your after-state improves lead time but leaves rework unchanged. Would you accept, revise, or reject the change? Defend your decision using two criteria.

---

**5.** (Analyze) Audit the software-produced 21-order comparison. Confirm that its cohort, boundary, clock, physical unit, percentile convention, and promise denominator match the baseline. Verify the after-state violation count directly from the ready timestamps, then write a short Check summary that also notes the supplied all-order PCE result and unchanged correction count.

---

**6.** (Evaluate) Before opening the answer, make a provisional Check/Act recommendation from this Friday. What would you retain for another test, what would you monitor, and what can you not yet attribute to 5S? Then inspect the calendar-year simulation evidence in the answer and explain whether it changes your recommendation.

---

**7.** (Apply) Prepare a six-bullet owner brief for your proposed improvement using the categories above. End with one difficult question you would want the owner or a process participant to ask before authorizing the trial.

### 12.5.1. Journal and Reflect

Which part of your original proposal changed after you inspected the evidence? What result did you find most persuasive, and what uncertainty still limits the recommendation?

Experimentation and management both require some tolerance for ambiguity. Describe a time when you had to make a responsible decision before the evidence clearly favored one explanation. How could you remain open to several plausible explanations without allowing uncertainty to become indecision? Name one practice that would help you take a concrete next step without pretending to know more than you do.

The guided cycle is now complete. The capstone asks you to reconstruct the same reasoning with a process of your own.

## 12.6. Chapter Summary

- Check compares the after state with the baseline using the same boundary, cohort, units, and calculation method.
- One favorable result supports a provisional judgment; repeated comparable observations show whether the pattern survives ordinary variation.
- Descriptive evidence does not by itself quantify uncertainty or prove that an intervention caused a real-world effect; later statistical methods strengthen those judgments.
- Act states what to retain, revise, monitor, and investigate, then carries the remaining gap into the next PDCA cycle.
- A useful owner presentation connects customer value, operating mechanism, business tradeoffs, evidence limits, and a specific decision request.

# 13.

# White Belt Capstone: Run the Cycle

## 13.1. Why This Chapter Matters

You have now completed one guided PDCA cycle from an ordinary customer complaint to a measured Act recommendation. The capstone asks whether you can carry that reasoning into a process whose answer the book has not supplied. Your work need not produce a dramatic success. It must make the problem, evidence, intervention, result, and next decision visible enough for another person to examine.

## 13.2. Learning Objectives

By the end of this chapter, the student will be able to:

- Leading and Managing Projects Complete and communicate an evidence-based PDCA cycle on a small daily home process, including what improved, what did not, and what should happen next **Apply**

The Wash n' Fold case demonstrated the whole sequence with guidance: frame the customer problem, observe and map the work, establish a baseline, diagnose supported waste, design and test a bounded intervention, compare the result, and make an Act decision. Now you will build the same chain of reasoning from your own evidence. The value of the capstone lies in the integrity of that chain, not in arranging for every measure to improve.

## 13.3. Capstone Assignment

### 13.3.1. Apply the Full Cycle to Your Own Process

Choose something at home that you do every day and control well enough to observe and change safely. Making breakfast, packing a work bag, feeding a pet, preparing a coffee station, or resetting a room are suitable in scale, although you should choose your own process. Keep the boundary narrow enough that you can observe several complete runs without turning the capstone into a household renovation. Name who receives the result of the process, even when that customer is you, and state what that person values about it. Your goal is to demonstrate the same disciplined cycle you used on Wash n' Fold:

1. define the problem and set a SMART goal,
2. map the current state with an SPPM,
3. baseline the key metrics,
4. diagnose supported waste and identify what remains uncertain,
5. test one small countermeasure in the real process,
6. compare before-and-after evidence, including what did not improve, and make a recommendation.

Observe at least five complete comparable runs before the change and five after it, following the learning minimum in Chapter 7. If the process occurs once each day, the project will therefore take at least ten days of elapsed time, although the added project work should remain modest. That minimum lets you practice comparison; it does not by itself establish process stability or statistical significance. If another person participates in the process, ask them to check your current-state map and discuss the proposed change before the trial. If the change cannot be tested safely or within your authority, narrow the change or choose another daily process.

You may use the Mapping Worksheet Set or equivalent software to calculate repeated durations and summaries from the observations you enter, or complete the small dataset with the printed worksheets and elementary arithmetic. If you use software, verify at least one activity record's P/T, W/T, and S/T and one complete run's lead time by hand so you can check what the software is doing. You remain responsible for the units, boundary, aggregation, denominator, and interpretation whichever tool you choose.

Assemble the work as one integrated Capstone Project Packet in a single folder or document of your choice. If you are taking a course, follow any packet format your instructor assigns. The Mapping Worksheet Set supplies the timing and map-summary sheets; it is one part of the packet, not a workbook for the entire project. The sections below are parts of that packet,

not separate projects. Use the mapping and data collection file guide to choose the blank event log, Mapping Worksheet Set, Participant Verification Record, and Baseline Performance Readout. The completed Wash n' Fold copies show what each artifact preserves; fill the blank copies with your own process's observations and keep its boundary and clock consistent before and after the trial.

**!** Capstone Project Packet

Submit one packet containing the following sections from your own process:

- **Plan:** one clear problem statement with customer impact and one SMART goal.
- **Current-state evidence:** one event-log excerpt, its completed Mapping Worksheet Set, a verified current-state SPPM with its Participant Verification Record, and a Baseline Performance Readout across at least five runs.
- **Diagnosis and Trial Proposal:** one evidence-linked Eight Wastes diagnosis and one bounded countermeasure with its mechanism, risks, owner, measures, guardrails, and decision rules; include one-page standard work only when the change creates or alters a repeatable operating method.
- **After-state evidence:** one observed dataset containing at least five complete comparable runs, recorded under definitions comparable to the baseline.
- **Check and Act:** one before-and-after comparison stating what improved and what did not, followed by a short recommendation to standardize, revise, abandon, or retest the intervention.

The baseline readout reports lead time, VA time, time outside VA, and PCE, plus Takt, WIP, or a Little's Law relationship when that measure has a sensible interpretation for the chosen process. Distinguish RNVA from NVA within that remainder where your observations support the classification. If software performs the repetitive arithmetic, the packet includes the two manual spot-checks described above. If you work entirely by hand, the corresponding worksheet calculations satisfy that requirement.

Success at White Belt level is not a favorable result. It is a transparent, measurable improvement cycle that shows you can transfer the method from the worked case to a real process.

> **Capstone Rubric (White Belt)**
>
> Use this rubric to evaluate your capstone submission.

| Criterion | Weight | Proficient (Meets White Belt Standard) |
|---|---|---|
| Problem definition and SMART goal quality | 15% | Problem is specific, evidence-based, and linked to customer impact; SMART goal is measurable and time-bound. |
| Current-state SPPM quality | 15% | Map has clear start/end, coherent step sequence, valid VA/RNVA/NVA tagging, and timings. |
| Baseline metric correctness | 15% | Lead time, VA time, time outside VA, and PCE are computed and interpreted correctly; RNVA and NVA remain distinct where the evidence supports it. Takt, WIP, or Little's Law is included when it fits the chosen process rather than being forced onto it. |
| Waste diagnosis and countermeasure design | 20% | Waste claims match the evidence; the intervention has a plausible mechanism, constraints, and measures. |
| After-state and before-and-after comparison | 20% | The real trial is documented, comparisons use consistent definitions, and the analysis states what improved and what did not. |

| | | |
|---|---|---|
| Check/Act judgment | 15% | Recommendation follows from the evidence, addresses risks and remaining uncertainty, and defines the next action. |

Suggested interpretation:

- **90–100%:** excellent-quality White Belt work,
- **80–89%:** strong and complete,
- **70–79%:** acceptable with targeted revision,
- **<70%:** rework needed before sign-off.

### 13.3.2. Review and Revise Your Packet

For each rubric criterion, point to the observation, calculation, or decision in your packet that supports it. Identify the weakest criterion and revise that part before calling the packet complete. If the gap requires new observations rather than a better explanation, record what you still need to collect and make your recommendation provisional. When another person is available, ask them to trace your recommendation back to the evidence and record any disagreement and your response. Independent learners can use this review themselves; an instructor's sign-off is required only when their course specifies it. Completing this practice packet does not itself award a credential or replace the separate LMS assessment.

## 13.4. Capstone Crosswalk

This crosswalk shows how the Wash n' Fold artifacts you completed chapter by chapter serve as models for the corresponding parts of your own capstone.

| Prior chapter artifact | Where it appears in capstone deliverable |
|---|---|
| Problem statement + SMART goal (PDCA Plan) | Problem definition and target-condition section |
| Mapping boundary + current-state SPPM | Baseline process documentation section |
| Baseline Performance Readout (operating L/T, PCE, and a fitting flow measure) | Before-state quantitative baseline section |

| Prior chapter artifact | Where it appears in capstone deliverable |
|---|---|
| Eight Wastes diagnosis | Evidence-linked improvement target |
| Unified Trial Proposal | Countermeasure, mechanism, ownership, measurement plan, guardrails, and decision rules |
| Guided Check/Act evaluation | Before-and-after comparison and Act recommendation |

### 13.5. What Counts as a Defensible Result

A successful capstone does not require a successful intervention. An honest finding that the chosen change did not improve the customer outcome can demonstrate excellent White Belt reasoning if the measures are consistent and the next recommendation responds to the evidence. An unexpected result may mean the intervention was poorly chosen, another problem still controls performance, the process had little room to improve under its current conditions, the method was not carried out consistently, or ordinary variation obscured the effect. White Belt evidence may not distinguish among those explanations, so state them as possibilities and identify the next observation or test instead of declaring one to be true.

Do not hide adverse effects in order to produce a cleaner story. If lead time falls while errors rise, both facts belong in the recommendation. If a local setup measure improves while the customer promise does not, report the local gain and investigate the remaining system delay. If the pilot is too small or too short to support confidence, describe the result as provisional and define the additional observation needed.

Later belts will introduce statistical methods for measuring uncertainty and evaluating whether an apparent improvement is distinguishable from ordinary variation. At White Belt, preserve comparable observations, identify the limits of the evidence, and avoid treating one favorable result as proof.

### 13.6. Exercises

*Answers and guidance for these exercises are collected in the solutions appendix.*

**1.** (Analyze) Review your capstone artifact set. Identify its weakest evidence link and one assumption in your recommendation that could fail in practice. State what observation or test would strengthen that link.

**2.** (Evaluate) Evaluate your metric set for blind spots. Name one important customer or process outcome currently unmeasured and explain how you would measure it next cycle.

---

**3.** (Apply) Write a two-sentence executive summary for your capstone that includes the customer impact, evidence status, and next action.

---

**4.** (Analyze) Compare two recommendations: one with high impact and high effort, the other with moderate impact and low effort. Which would you stage first in Act, and what evidence supports that sequence?

---

**5.** (Evaluate) A teammate argues that documenting what did not improve will weaken stakeholder confidence. Evaluate this claim and justify your response.

---

**6.** (Apply) Identify a remaining problem supported by your capstone evidence and write a SMART goal for the next PDCA cycle.

### 13.6.1. Journal and Reflect

Write a short note on what you learned about your own improvement habits from this capstone and what you would pay closer attention to the next time you run a cycle.

### 13.6.2. Reflection Question

Write three or four sentences on one moment in your capstone where the evidence contradicted your initial intuition and how that changed your decision.

The capstone ends one cycle by opening the next. Carry its unresolved gap forward rather than treating submission as the end of improvement.

### 13.7. Chapter Summary

- The capstone asks you to construct and defend a complete PDCA cycle on a small daily home process after completing the guided Wash n' Fold cycle.
- A coherent improvement argument connects the customer problem, current-state evidence, waste diagnosis, intervention mechanism, after-state comparison, and Act decision.
- An intervention does not have to succeed for the capstone to succeed; honest negative and mixed results can support strong decisions.
- White Belt evidence should be comparable and transparent about its limits, leaving rigorous statistical treatment of uncertainty for later study.

### 13.8. Further Reading

A useful historical complement is this short clip of Steve Jobs discussing the quality movement of the 1980s:

**Steve Jobs on Quality**

Video available online

https://youtu.be/kib6uXQsxBA

Use the written activity below when offline.

**Media Credit**

Source: Steve Jobs on Quality, YouTube. Speaker: Steve Jobs; uploader: Dawn Lindsley. The uploader is not identified here as the original recording's creator or rights holder. Rights: reference link and platform-provided embed only; no recording, stills, or transcript are redistributed or relicensed by this book.

**Written alternative.** This optional clip is not needed to complete the capstone. Consider this original discussion prompt instead: a manager an-

nounces a commitment to quality but leaves the release rule, evidence collection, and review responsibilities unchanged. What would distinguish a changed process from a changed slogan? Use your Check/Act recommendation to identify one observable difference.

Part III.

# Readings

# 14.

# The Student, the Fish and Agassiz

**Reading Credit**

Samuel H. Scudder, "In the Laboratory with Agassiz", *Scientific American* 30, no. 16 (April 18, 1874), p. 244. DOI: 10.1038/scientificamerican04181874-244.
Rights: Historical text is public domain in the United States (published before 1931); used under public-domain terms, not fair use. The photograph below has its own separate license.

It was more than fifteen years ago that I entered the laboratory of Professor Agassiz, and told him I had enrolled my name in the scientific school as a student of natural history. He asked me a few questions about my object in coming, my antecedents generally, the mode in which I afterwards proposed to use the knowledge I might acquire, and finally, whether I wished to study any special branch.

To the latter I replied that while I wished to be well grounded in all departments of zoology, I purposed to devote myself specially to insects.

"When do you wish to begin?" he asked.

"Now," I replied.

This seemed to please him, and with an energetic "Very well," he reached from a shelf a huge jar of specimens in yellow alcohol. "Take this fish," he said, "and look at it; we call it a Haemulon; by and by I will ask what you have seen." With that he left me, but in a moment returned with explicit instructions as to the care of the object entrusted to me. "No man is fit to be a naturalist," said he, "who does not know how to take care of specimens."

I was to keep the fish before me in a tin tray, and occasionally moisten the surface with alcohol from the jar, always taking care to replace the

stopper tightly. Those were not the days of ground glass stoppers, and elegantly shaped exhibition jars; all the old students will recall the huge, neckless glass bottles with their leaky, wax-besmeared corks, half-eaten by insects and begrimed with cellar dust. Entomology was a cleaner science than ichthyology, but the example of the professor who had unhesitatingly plunged to the bottom of the jar to produce the fish was infectious; and though this alcohol had "a very ancient and fish-like smell," I really dared not show any aversion within these sacred precincts, and treated the alcohol as though it were pure water. Still I was conscious of a passing feeling of disappointment, for gazing at a fish did not commend itself to an ardent entomologist. My friends at home, too, were annoyed, when they discovered that no amount of eau de cologne would drown the perfume which haunted me like a shadow.

In ten minutes I had seen all that could be seen in that fish, and started in search of the professor, who had, however, left the museum; and when I returned, after lingering over some of the odd animals stored in the upper apartment, my specimen was dry all over. I dashed the fluid over the fish as if to resuscitate it from a fainting-fit, and looked with anxiety for a return of a normal, sloppy appearance. This little excitement over, nothing was to be done but return to a steadfast gaze at my mute companion. Half an hour passed, an hour, another hour; the fish began to look loathsome. I turned it over and around; looked it in the face—ghastly; from behind, beneath, above, sideways, at a three-quarters view—just as ghastly. I was in despair; at an early hour, I concluded that lunch was necessary; so with infinite relief, the fish was carefully replaced in the jar, and for an hour I was free. On my return, I learned that Professor Agassiz had been at the museum, but had gone and would not return for several hours. My fellow students were too busy to be disturbed by continued conversation. Slowly I drew forth that hideous fish, and with a feeling of desperation again looked at it. I might not use a magnifying glass; instruments of all kinds were interdicted. My two hands, my two eyes, and the fish; it seemed a most limited field. I pushed my fingers down its throat to see how sharp its teeth were. I began to count the scales in the different rows until I was convinced that that was nonsense. At last a happy thought struck me—I would draw the fish; and now with surprise I began to discover new features in the creature. Just then the professor returned.

"That is right," said he, "a pencil is one of the best eyes. I am glad to notice, too, that you keep your specimen wet and your bottle corked." With these encouraging words he added— "Well, what is it like?"

He listened attentively to my brief rehearsal of the structure of parts whose names were still unknown to me; the fringed gill-arches and movable operculum; the pores of the head, fleshly lips, and lidless eyes; the lateral line,

the spinous fin, and forked tail; the compressed and arched body. When I had finished, he waited as if expecting more, and then, with an air of disappointment: "You have not looked very carefully; why," he continued, more earnestly, "you haven't seen one of the most conspicuous features of the animal, which is as plainly before your eyes as the fish itself. Look again; look again!" And he left me to my misery.

I was piqued; I was mortified. Still more of that wretched fish? But now I set myself to the task with a will, and discovered one new thing after another, until I saw how just the professor's criticism had been. The afternoon passed quickly, and when, towards its close, the professor inquired, "Do you see it yet?"

"No," I replied. "I am certain I do not, but I see how little I saw before."

"That is next best," said he earnestly, "but I won't hear you now; put away your fish and go home; perhaps you will be ready with a better answer in the morning. I will examine you before you look at the fish."

This was disconcerting; not only must I think of my fish all night, studying, without the object before me, what this unknown but most visible feature might be, but also, without reviewing my new discoveries, I must give an exact account of them the next day. I had a bad memory; so I walked home by Charles River in a distracted state, with my two perplexities. The cordial greeting from the professor the next morning was reassuring; here was a man who seemed to be quite as anxious as I that I should see for myself what he saw.

"Do you perhaps mean," I asked, "that the fish has symmetrical sides with paired organs?"

His thoroughly pleased, "Of course, of course!" repaid the wakeful hours of the previous night. After he had discoursed most happily and enthusiastically—as he always did—upon the importance of this point, I ventured to ask what I should do next. "Oh, look at your fish!" he said, and left me again to my own devices. In a little more than an hour he returned and heard my new catalogue. "That is good, that is good!" he repeated, "but that is not all; go on." And so for three long days, he placed that fish before my eyes, forbidding me to look at anything else, or to use any artificial aid. "Look, look, look," was his repeated injunction.

This was the best entomological lesson I ever had—a lesson whose influence was extended to the details of every subsequent study; a legacy the professor has left to me, as he left it to many others, of inestimable value, which we could not buy, with which we cannot part. A year afterwards, some of us were amusing ourselves with chalking outlandish beasts upon the blackboard. We drew prancing star-fishes; frogs in mortal combat; hydro-headed worms;

stately craw-fishes, standing on their tails, bearing aloft umbrellas; and grotesque fishes, with gaping mouths and staring eyes. The professor came in shortly after, and was as much amused as any at our experiments. He looked at the fishes.

"Haemulons, every one of them," he said; "Mr. [Scudder] drew them."

True; and to this day, if I attempt a fish, I can draw nothing but Haemulons.

The fourth day a second fish of the same group was placed beside the first, and I was bidden to point out the resemblances and differences between the two; another and another followed, until the entire family lay before me, and a whole legion of jars covered the table and surrounding shelves; the odor had become a pleasant perfume; and even now, the sight of an old six-inch worm-eaten cork brings fragrant memories! The whole group of Haemulons was thus brought into review; and whether engaged upon the dissection of the internal organs, preparation and examination of the bony framework, or the description of the various parts, Agassiz's training in the method of observing facts in their orderly arrangement, was ever accompanied by the urgent exhortation not to be content with them.

"Facts are stupid things," he would say, "until brought into connection with some general law."

At the end of eight months, it was almost with reluctance that I left these friends and turned to insects; but what I gained by this outside experience has been of greater value than years of later investigation in my favorite groups.

**Figure 14.1.:** Haemulon sciurus

**Media Credit**

Source: File: Blue Stripe Grunt. Haemulon sciurus, Wikimedia Commons. Creator: Brian Gratwicke.
Rights: CC BY 2.0. Used under license (not fair use).
Modification: No substantive modification; resized by the publication renderer.

**Part IV.**

# Appendices

# 15.

# Lean Six Sigma Learning Map

This appendix maps the capabilities promised by this curriculum across five belt levels and six knowledge domains. It is both a planning document and an audit standard. If the map claims that a learner should be able to perform a task by the end of a belt, the curriculum should teach that capability, provide an opportunity to practice it, and assess it at the stated level.

The map is intentionally coarse-grained. Its job is to show stable capability bands that matter for belt-level planning, sequencing, assessment, and role expectations, not to reproduce every chapter-level Learning Objective one for one. Additional detail should be added only when multiple chapters, an entire belt, or a recurring assessment decision depends upon it.

### 15.1. How to Read the Map

Each populated cell gives the highest level of cognitive performance expected by the end of that belt:

- **Remember:** recall relevant facts, terms, formulas, or procedures.
- **Understand:** explain a concept, recognize examples, or interpret a simple representation.
- **Apply:** use the capability in a concrete situation appropriate to the belt.
- **Analyze:** distinguish relevant features, investigate relationships, diagnose problems, or compare alternatives.
- **Evaluate:** judge the quality, suitability, consequences, or limitations of an approach and defend that judgment.
- **Create:** integrate capabilities to design a new analysis, model, intervention, or system.

Lower-level performances are normally implied by a higher-level expectation. For example, `Analyze` presumes that the learner can also remember,

understand, and apply the relevant material. An em dash (—) means that the curriculum makes no assessed end-of-belt promise for that capability.

The cells describe performance within the scope and complexity appropriate to each belt. `Apply` at White Belt may mean constructing a simple process map or problem statement independently, while `Apply` at Black Belt may involve a substantially more technical method and an ambiguous organizational setting. The levels are a practical vocabulary, not a mechanical ladder; some capabilities emphasize evaluation while others culminate in creation.

The cognitive levels and the educational philosophy behind the map are explained in The Method and the Madness.

### 15.2. Belt Levels

- **White Belt (WB):** orientation plus limited independent use of foundational process-improvement concepts.
- **Yellow Belt (YB):** practical application of fundamental process, data, and computational tools.
- **Green Belt (GB):** independent project execution and intermediate analytical capability.
- **Black Belt (BB):** advanced analytical work, model and tool development, and coaching responsibility.
- **Master Black Belt (MBB):** evaluation of technical work, enterprise program design, and strategic governance.

At Master Black Belt, `Evaluate` in a technical row means that the practitioner can assess the appropriateness, quality, risks, and organizational implications of the work. It does not imply that every Master Black Belt must remain a cutting-edge specialist in every advanced method.

### 15.3. Knowledge Domains

- Lean Six Sigma Principles and Tools
- Leading and Managing Projects
- Data Understanding
- Computational and Data Systems
- Statistical Analysis
- Business Understanding

These are practical curriculum-design categories rather than rigid disciplinary boundaries. Many capabilities could reasonably appear in more

than one domain; each is placed where it is most useful for sequencing and assessment.

## 15.4. Domain-by-Belt Crosswalk

### 15.4.1. Lean Six Sigma Principles and Tools

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Five Principles of Lean | Understand | Apply | Analyze | Evaluate | Evaluate |
| Respect for People and Kaizen Culture | Understand | Apply | Analyze | Evaluate | Evaluate |
| Eight Wastes | Analyze | Analyze | Analyze | Evaluate | Evaluate |
| 5S | Apply | Apply | Analyze | Evaluate | Evaluate |
| PDCA Framework | Apply | Analyze | Evaluate | Create | Evaluate |
| A3 Framework | — | Apply | Analyze | Create | Evaluate |
| DMAIC Framework | — | — | Apply | Create | Evaluate |
| Basic Process Mapping | Apply | Analyze | Analyze | Create | Evaluate |
| FLO Process Modeling | — | Apply | Analyze | Create | Evaluate |
| Rendered Process Representations (including SPPM and VSM) | Apply | Apply | Analyze | Create | Evaluate |
| Spaghetti Diagrams, SIPOC, and IPO Analysis | — | Apply | Analyze | Evaluate | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Seven Basic Quality Tools | — | Apply | Analyze | Evaluate | Evaluate |
| Process Metrics (P/T, C/T, L/T, PCE, FPY, RTY, and Percentiles) | Apply | Analyze | Analyze | Evaluate | Evaluate |
| Control-Chart Interpretation | — | Understand | Analyze | Evaluate | Evaluate |
| Control-Chart Construction and Selection | — | — | Apply | Evaluate | Evaluate |
| Critical-to-Quality Characteristics and Kano Analysis | — | Apply | Analyze | Evaluate | Evaluate |
| Customer Value and the Value of Lean Six Sigma | Understand | Apply | Analyze | Evaluate | Evaluate |

White Belt uses PDCA to organize its first complete improvement cycle. A3 begins at Yellow Belt, where the learner can apply the reporting and problem-solving structure to bounded work. DMAIC begins at Green Belt, alongside the stronger project, data, and statistical foundation needed to use it responsibly.

FLO is the declarative source language used to implement process descriptions in this curriculum. SPPM, value-stream maps, and other visual forms are treated as representations that can be generated from or aligned with the underlying process model. White Belts are not expected to write FLO, but they should be able to construct and interpret a basic process map. Yellow Belts begin implementing models in FLO; Green Belts analyze and elaborate them; Black Belts may extend the models, tooling, or language when the work requires it.

### 15.4.2. Leading and Managing Projects

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Problem Statements and SMART Goals | Apply | Analyze | Evaluate | Create | Evaluate |
| Project Selection and Strategic Alignment | — | Apply | Analyze | Evaluate | Evaluate |
| Project Scope and Level of Effort | — | Understand | Analyze | Evaluate | Evaluate |
| Business-Case Construction | — | Apply | Analyze | Evaluate | Evaluate |
| Meeting Agendas, Decisions, and Notes | — | Apply | Analyze | Evaluate | Evaluate |
| Project Charters | — | — | Apply | Evaluate | Evaluate |
| Project-Planning and Management Tools | — | — | Apply | Analyze | Evaluate |
| Team Selection and Role Definition | — | Understand | Apply | Evaluate | Evaluate |
| Running Effective Meetings | — | Understand | Apply | Evaluate | Evaluate |
| Facilitation | — | Understand | Apply | Analyze | Evaluate |
| Stakeholder Engagement and Communication | Apply | Apply | Analyze | Evaluate | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Responsible Improvement and Ethical Judgment | Understand | Apply | Analyze | Evaluate | Evaluate |
| Evaluating Unintended Consequences and Stakeholder Impacts | Understand | Apply | Analyze | Evaluate | Evaluate |
| Reflective Practice and Learning from Failure | Apply | Apply | Analyze | Evaluate | Evaluate |
| Coaching and Developing Practitioners | — | — | Understand | Apply | Evaluate |
| Project-Portfolio Governance | — | — | Understand | Analyze | Evaluate |

Responsible improvement is treated as a professional capability rather than a separate code of conduct. Learners are expected to examine assumptions and incentives, consider who bears the risks and receives the benefits of a change, seek the knowledge of people closest to the work, and exercise judgment proportionate to the stakes.

### 15.4.3. Data Understanding

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Structured Process Observation and Data Capture | Apply | Analyze | Analyze | Evaluate | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Baseline Construction from Process Data | Apply | Analyze | Analyze | Evaluate | Evaluate |
| Obtaining and Importing Datasets | — | Apply | Analyze | Evaluate | Evaluate |
| Basic Data Cleaning and Validation | — | Apply | Analyze | Evaluate | Evaluate |
| Missing-Data and Data-Quality Strategies | — | Understand | Analyze | Evaluate | Evaluate |
| Levels of Measurement and Data Types | — | Apply | Analyze | Evaluate | Evaluate |
| Summarizing and Grouping Data | — | Apply | Analyze | Evaluate | Evaluate |
| Reshaping Data (Pivots, Melts, and Group Operations) | — | Apply | Analyze | Evaluate | Evaluate |
| Identifying and Handling Outliers | — | Understand | Analyze | Evaluate | Evaluate |
| Data Visualization | — | Apply | Analyze | Create | Evaluate |
| Exploratory Data Analysis | — | Apply | Analyze | Evaluate | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Time-Series Display and Graphical Interpretation | — | Apply | Analyze | Evaluate | Evaluate |
| Formal Time-Series Analysis | — | — | Apply | Analyze | Evaluate |
| Qualitative Data Collection and Analysis | — | Understand | Apply | Analyze | Evaluate |
| Geographical Data | — | — | — | Apply | Evaluate |
| Machine Learning | — | — | — | Apply | Evaluate |
| Dimensionality Reduction (PCA, Factor Analysis, t-SNE, and UMAP) | — | — | — | Apply | Evaluate |
| Feature Selection and Feature Extraction | — | — | — | Apply | Evaluate |

By the end of Yellow Belt, learners should be able to obtain a prepared dataset and do something meaningful with it in Python: import it, inspect it, perform basic cleaning and reshaping, summarize it, visualize it, and interpret the result. Green Belt develops the capability to work more independently with messy organizational data and make defensible handling decisions.

#### 15.4.4. Computational and Data Systems

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Computer and File-System Fundamentals | — | Apply | Analyze | Evaluate | Evaluate |
| Basic Terminal Navigation and Commands | — | Apply | Analyze | Evaluate | Evaluate |
| Shell Workflows and Command-Line Tooling | — | Understand | Apply | Create | Evaluate |
| Jupyter Notebook Workflows | — | Apply | Analyze | Create | Evaluate |
| Python Programming for Process Analysis | — | Apply | Analyze | Create | Evaluate |
| Importing and Using Python Packages | — | Apply | Analyze | Create | Evaluate |
| Reproducible Notebook and Script Workflows | — | Apply | Analyze | Create | Evaluate |
| Dependency and Environment Management | — | Understand | Apply | Analyze | Evaluate |
| Data Pipelines and Workflow Composition | — | — | Apply | Create | Evaluate |
| Workflow Automation | — | — | Apply | Create | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| SQL Querying and Relational Data Modeling | — | — | Apply | Analyze | Evaluate |
| Version Control and Collaborative Development | — | — | Apply | Analyze | Evaluate |
| Testing and Validation of Analysis Code | — | — | Apply | Analyze | Evaluate |
| Reusable Packages and Internal Tooling | — | — | Apply | Create | Evaluate |
| Discrete-Event Simulation with SimPy | — | — | — | Create | Evaluate |
| Optimization and Operations-Research Workflows | — | — | — | Apply | Evaluate |
| Process Mining and Event-Log Workflows | — | — | — | Apply | Evaluate |

Yellow Belt computing is deliberately modest but real: Jupyter notebooks, basic terminal commands, introductory Python, package use, and reproducible handling of prepared datasets. Green Belt adds substantial command-line work, scripts, environments, data pipelines, SQL, version control, testing, and automation. Black Belt uses that foundation to construct simulations, advanced workflows, reusable tools, and extensions when an existing tool is inadequate.

### 15.4.5. Statistical Analysis

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Descriptive Statistics | Apply | Apply | Analyze | Evaluate | Evaluate |
| Distribution Concepts, the Standard Normal Distribution, and Percentiles | — | Apply | Analyze | Evaluate | Evaluate |
| Scientific and Experimental Reasoning | Analyze | Analyze | Evaluate | Create | Evaluate |
| Hypothesis-Testing Foundations (including the Lady Tasting Tea) | — | Apply | Analyze | Evaluate | Evaluate |
| Student's *t*-Tests | — | Apply | Analyze | Evaluate | Evaluate |
| Confidence Intervals and Sample-Size Concepts | — | Apply | Analyze | Evaluate | Evaluate |
| Type I and Type II Errors | — | Understand | Analyze | Evaluate | Evaluate |
| Scatterplots and Linear Association | — | Apply | Analyze | Evaluate | Evaluate |
| Common Probability Distributions | — | Understand | Apply | Analyze | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Distribution Assessment and Assumption Checking | — | — | Apply | Analyze | Evaluate |
| Linear Regression | — | — | Apply | Analyze | Evaluate |
| Analysis of Variance (ANOVA) | — | — | Apply | Analyze | Evaluate |
| Multivariate Analysis of Variance (MANOVA) | — | — | — | Apply | Evaluate |
| Measuremen System Analysis | — | — | Apply | Analyze | Evaluate |
| Design of Experiments | — | — | Apply | Create | Evaluate |
| Survey Design and Analysis | — | — | Apply | Analyze | Evaluate |

Yellow Belt introduces formal statistical reasoning through descriptive statistics, the concept of a distribution, confidence intervals, the Lady Tasting Tea, and Student's *t*-test. The emphasis is on interpreting evidence and understanding what the methods establish, not accumulating a large catalog of tests. Green Belt broadens inferential capability and adds independent assumption checking, regression, ANOVA, measurement-system analysis, and experimental design.

### 15.4.6. Business Understanding

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Customer and Stakeholder Value | Apply | Apply | Analyze | Evaluate | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Financial Statements from a Process-Improvemen Perspective | — | Under-stand | Apply | Analyze | Evaluate |
| Fundamen-tals of Accounting | — | Under-stand | Apply | Analyze | Evaluate |
| Time Value of Money | — | Apply | Analyze | Evaluate | Evaluate |
| Return on Investment | — | Apply | Analyze | Evaluate | Evaluate |
| Cost of Quality and Cost of Poor Quality | — | Apply | Analyze | Evaluate | Evaluate |
| Strategic Alignment and Mission Fit | — | Apply | Analyze | Evaluate | Evaluate |
| Benefits Realization and Sus-tainment | Apply | Apply | Analyze | Evaluate | Evaluate |
| Competi-tive Advantage and Strategic Planning | — | Under-stand | Analyze | Evaluate | Evaluate |
| Balanced Scorecard | — | — | Apply | Analyze | Evaluate |
| ISO 9001:2015 Quality-Management Principles | — | — | Under-stand | Apply | Evaluate |
| Enterprise Improve-ment Gover-nance | — | — | Under-stand | Analyze | Evaluate |

| Capability | WB | YB | GB | BB | MBB |
|---|---|---|---|---|---|
| Organizational Capability and Curriculum Design | — | — | Understand | Analyze | Create |

The business domain keeps improvement work tied to customer value, organizational purpose, financial reality, and the capacity to sustain gains. Yellow Belts learn enough financial and strategic framing to support small improvement efforts. Green and Black Belts perform increasingly consequential analysis, while Master Black Belts evaluate enterprise decisions and create systems for organizational capability development.

### 15.5. Auditing the Curriculum Against the Map

Because the map is a curriculum promise, every populated cell should be traceable to evidence in the curriculum. For each capability and belt, an audit should be able to identify:

1. One or more Learning Objectives stating the promised performance;
2. Instruction and examples sufficient to support that performance;
3. Practice at an appropriate level of complexity;
4. Feedback or answer guidance that helps learners improve; and
5. An assessment or project artifact that supplies evidence of the end-of-belt capability.

A missing link is information about the curriculum. If a capability is promised but not taught, practiced, or assessed, the curriculum should be revised or the promise should be removed. If learners repeatedly fail to demonstrate a mapped capability, that pattern should trigger investigation of the instruction, prerequisite sequence, exercises, assessment, and stated belt placement.

Use the map as a guide, not a rigid doctrine. Real projects often require moving backward to foundations or reaching forward for a more advanced tool. The map describes the capabilities the curriculum promises by each milestone; it does not prescribe the order in which every real problem must be solved.

# 16.

# The First Reckoning: What Did You Make One?

Arithmetic with Nisaba

### 16.1. Before You Begin

This appendix is for people who have learned arithmetic before, people who were once told they had learned it, and people who have spent years arranging their lives so the subject would not notice them.

The story that follows attempts to teach rather than merely summarize. Calculations appear when the characters need them to count, allocate, or challenge an account. After the coda, four "reckonings" name the ideas the characters discover, work through the arithmetic, and provide checks for your own understanding. Confident readers may skim that review and return when they need it. Readers rebuilding their confidence may find it better to work through it in order. Nothing in this chapter rewards speed.

By the end, you should be able to:

- distinguish a thing, a quantity, a number, a numeral, and a unit;
- compare quantities with equality and inequality after checking their units and boundaries;
- explain place value in decimal and sexagesimal notation, including zero as a count and as a placeholder;
- interpret addition, subtraction, multiplication, and division in context;

- use signed quantities to describe change without confusing a change with a final amount;
- connect fractions, decimals, ratios, rates, proportions, and percentages;
- distinguish exact results from useful approximations;
- estimate before using a machine and check whether its result answers the intended question;
- interpret exponents as repeated multiplication; and
- factor a positive integer into primes and state what the Fundamental Theorem of Arithmetic guarantees.

## 16.2. Act I: Six Circles of Stone

### 16.2.1. The spring reckoning

At first light the sheep came through the western gate of Reed Gate in a river of dust.

The settlement stood on raised ground between a branch of the canal and a mud-brick storehouse larger than any of its houses. The gate that gave Reed Gate its name had been rebuilt from bundled reeds so many times that no one knew which gate had named the place. The storehouse was different. Its thick walls, plastered bins, sealed jars, and shaded loading yard belonged to a temple household in Nippur. Barley obligations from the surrounding fields entered there; seed, rations, and measured advances left under seal. Wool waited there for the road to the city. The people of Reed Gate did not own what stood behind its doors, but almost every household depended on something that passed through them.

Lamassatum already waited beneath the loading awning with an empty jar and a clay token for barley owed to her household. She counted the people ahead of her twice, although there were only three.

Re’u the shepherd entered the gate with more of the flock, then stood to one side with his sling looped at his belt and his waterskin hanging from the opposite shoulder. The morning was cold enough for the woolen cloak he had wrapped twice around himself before dawn, but the day would be hot before the last animal was through. Spring made promises in this country that summer collected with interest. This year the canal had already fallen from the dark line it usually held against the bank. Re’u had noticed. He did not yet know what the difference would mean.

The temple’s order named six groups of sixty for this flock. Re’u had helped select the animals from the district flocks before dawn. The gate count would establish whether all of them had arrived.

He moved one stone from his left hand to his right for every sheep that entered the fold. When his right hand filled, he dropped the stones into a shallow bowl and began again. Ten stones made a little mound. Six mounds made a circle. When a circle was complete, he set one larger stone beside it. The scribe Ili-iddinam called a group of six mounds *šūši*, sixty. Re'u usually just called it a circle.

The flock pressed forward. Re'u watched backs, ears, feet, and the dark seams between animals. The count could be lost in any of them. One ewe turned at the gate when her lamb cried behind her. Re'u caught the ewe with his knee, drew the lamb through with one hand, and did not move a stone until he had decided which animal he was counting.

Ili-iddinam, sat beneath a reed awning whittling a new stylus with a wet clay tablet across his knees. He had arrived from Nippur for the reckoning wearing a clean fringe and an expression that suggested the dust had breached an agreement. He marked each completed circle while Abum-ilum, overseer of the storehouse, watched the gate and the tablet in alternation.

Abum-ilum held the storehouse seal and answered for whatever entered or left its doors. Ili-iddinam was not his household servant. He wrote the account that would go back to Nippur, where a loss that could not be followed on clay might become Abum-ilum's loss whether or not he had caused it. The overseer could order a jar opened or an animal transferred. He could not make the tablet say that the jar remained sealed.

Lamassatum lifted her empty jar toward Abum-ilum. "Before the day grows hot?"

"After the flock is witnessed," he said.

Without looking up from the clay, Ili-iddinam added, "And after the storehouse seal is witnessed."

Abum-ilum looked at the young writer. "Yes. After that."

"Five," Ili-iddinam called after the fifth large stone.

"Five circles," Re'u said.

"Five sixties."

"Then write five sixties."

The young man looked wounded. "That is what I am writing."

Re'u let the sixth group pass and set down the sixth large stone. The remaining bowl was empty.

"And six *šūši* make three hundred sixty exactly," said Ili-iddinam.

Abum-ilum nodded. “Six sixties of sheep from the canal district entrusted to Abum-ilum at the spring reckoning.”

Re’u knew the number without needing the tablet. He knew more than the number. He knew which ewe favored her left hind leg, which ram would strike if approached from the front, which yearling had learned to nose open a reed hurdle, and which three animals would remain close to water even when better grass lay beyond the ridge. The tablet knew none of these things.

Erishtum came from the wool shelter carrying a basket of cords. She was the widow of Re’u’s older brother, Ilum-bani, and had remained in the household with her two young children. Her son, Ahu-waqar, approached every new task with grave determination and usually misplaced one necessary part of it. Her younger daughter, Amat-Sin, had not yet learned that an adult’s silence might be intended to end a conversation. The wool, the sealed jars, and the measures of barley passed through her hands often enough that Re’u no longer remembered when he had begun asking her before promising any of them. He occasionally remembered after promising them. Those were different conversations.

She counted the six large stones, then looked at the flock.

“All of them?”

“All that were entrusted,” Abum-ilum said.

Erishtum did not answer him. She asked Re’u, “All of them?”

“All.”

Only then did she carry the cords inside.

### 16.2.2. The bronze girl

The shouting began beyond the outer wall.

At first Re’u thought a dog had worried the flock. Then the sheep nearest the western gate moved away from it together, not in panic but with the grave unanimity animals sometimes displayed before a storm.

Something stood where the path met the wall.

It was shaped like a girl not yet grown to a woman’s height. Its garment fell in simple folds to its ankles, but the folds, the feet, the narrow hands, and the face were all one bronze body. There was no pin, seam, hinge, or opening that Re’u could see. No household he knew possessed so much bronze. No temple official would have allowed it to stand unattended in a field.

The figure stepped forward.

Bronze should not have moved that way. It should not have moved at all.

Abum-ilum dropped his staff. Ili-iddinam pressed both hands into his tablet and spoiled the last line. Someone behind Re'u said the name before the figure spoke.

"Nisaba."

The name moved through the people at the gate. Lady of grain. Mistress of the reed stylus. Keeper of accounts. Some knelt. Some backed away. Erishtum remained standing, although she tightened her grip on the basket.

The bronze figure looked from the people to the six large stones and the emptied bowl.

"Nisaba," she said.

Her mouth moved only a little. Her voice did not sound as though it had traveled through a bronze throat.

Abum-ilum recovered his staff and his office at approximately the same moment. "Lady," he said, "this flock belongs under the protection of the storehouse."

Nisaba looked at him.

"The flock," he added.

She turned back to the stones.

Re'u was relieved to discover that a divine visitation could also leave an overseer uncertain where to put his hands.

Nisaba crouched beside the six circles. She touched one of the larger stones, then one of the small ones. She separated three small stones from a mound, returned them, and touched the ewe nearest the gate.

"One," she said.

She touched another ewe.

"One."

She touched one stone.

"One."

Then she looked at Re'u.

He understood the question before she found words for it.

"One stone for one sheep," he said.

Nisaba placed a stone before the ewe. The ewe stepped around it. Nisaba moved another stone before another sheep. Soon a line of animals had passed while a line of stones accumulated in the dust.

“The stones are not sheep,” Ili-iddinam said.

Nisaba looked at him, waiting.

“But there is one for each,” he finished.

She set three stones in a row and waited until three sheep stood opposite them. With one finger she drew a line from each stone to one animal.

“Do the lines mean to say that they are the same?” Ili-iddinam asked.

“No,” Nisaba said after studying the lines. “Only that there are as many stones as sheep.”

It was the first complete answer she gave them.

### 16.2.3. The man beside the canal

By midmorning the lowest watering cut had clogged with reeds. Re’u left the crowd at the fold and followed the canal path with his staff across his shoulders. Sheep could wait for gods and officials to finish speaking. They could not wait long for water.

An old man sat where the path divided beside the cut. He wore a patched garment of no cut Re’u recognized. He was older than anyone Re’u knew how to place. His scalp was bare except for a dirty white fringe above his ears, and a mangy beard hung in thin knots from his chin. Both eyes were filmed white and fixed on nothing. A stick lay across his knees.

Before Re’u spoke, the man turned his blind face toward him.

“Re’u.”

Re’u stopped. “Who told you my name?”

“You did.”

“I have not.”

The old man considered this. “Then you have not done it yet.”

Re’u cleared the reeds from the cut while the stranger listened to the water begin moving again. When Re’u finished, the man held out one hand without asking where the path led. Re’u put that hand on his forearm.

“There is a bronze girl at the fold,” Re’u said.

The stranger’s grip tightened. “Yes.”

“You know her?”

“I remember her.”

Re’u led him back to Reed Gate. The man’s stick found the ground before his feet did, but he knew when the canal path narrowed and where the storehouse wall would return the sound of their steps.

At the fold, Re’u gave the introductions the stranger had not asked for. “Abum-ilum oversees the storehouse. Ili-iddinam writes its tablets. Erishtum is of my household and knows what the wool and jars contain better than I do.”

With each name, the old man’s face changed before the named person spoke. It was not surprise.

“You look as though you know us,” Erishtum said.

“I do.”

“We do not know you.”

“You will,” he said, and his smile faded.

He offered no name.

Nisaba turned toward him. The old man’s smile began before she spoke, as though he recognized even the length of her silence. It changed his whole ruined face.

“Nisaba,” he said.

“You know my name.”

“I have known you for a very long time, nearly all my long life, in fact.”

He raised one hand toward her, stopped before touching her, and closed it again around his stick. She watched the halted hand.

He walked to the stones without bowing. Re’u caught his elbow before his foot entered one of the circles. The old man accepted the correction without thanks or embarrassment.

He held out one hand toward the fold. Re’u guided it to a ewe’s shoulder.

“What is that?”

“One,” Nisaba said.

“One what?” asked the stranger.

She touched the animal. “Sheep.”

He crouched and searched the dust until his fingers found the stone Nisaba had paired with the ewe. “And that?”

“One.”

“One what?”

Nisaba looked from the stone to the sheep. “Stone.”

“The stone is not the sheep. The word is not the sheep. The mark is not the sheep. What have you counted?”

Nisaba touched the stone, then the ewe. “One beside one.”

“You have named the arrangement.”

The ewe moved to follow her lamb. The stone remained where Nisaba had placed it.

“The sheep,” Re’u said.

“Then how did the stones help?”

She followed the three lines she had drawn in the dust. “One stone for each sheep. No sheep twice. None omitted.”

The old man traced one of the lines with his stick. “Move this stone so that it claims the same animal as that one.”

Re’u moved it. “Now?”

“The stones now say that one animal was counted twice.”

“Stones are poor liars. Who made the pairing?”

Nisaba restored the lines. “I did.”

At the sound of a lamb, he extended his stick toward the ewe and the animal pressed against her side. “How many?”

“Two sheep.”

“Could they be one?”

Nisaba paused.

Re’u answered before she could. “One ewe and lamb if I am assigning a nursing pair. Two animals if I am counting mouths.”

The old man nodded. “The animals did not change. The count did because he changed what would count as one.”

He tapped the edge of the nearest circle with his stick.

Nisaba counted the circles, then touched one of the large stones at their centers. “Six groups. Sixty sheep in each.”

“Then what did you make one?”

This time the question did not sound foolish to Re'u. One sheep. One ewe-and-lamb pair. One flock. One stone standing for one animal. One mound of ten stones. One circle of sixty. The world had not supplied a single answer before the counting began.

"What I choose to count as one," Nisaba said.

"A unit," the stranger said.

The choice was hers and the word was his. Re'u understood both through the things she touched.

"And after you choose it?" the stranger asked.

"Count without repeating or omitting it."

"What have you counted?"

"What I made one."

The old man seemed pleased and terribly tired. "That is enough for a beginning."

Abum-ilum stepped closer. "Do you know what she is?"

The stranger turned his blind face toward Nisaba. "Before spring returns, the tablet writer will ask whether an account is correct. She will tell him another hand can follow it."

Ili-iddinam looked offended to have his future question answered in public. "How do you know what she will say?"

"I remember her saying it," the man said flatly.

"I did not say that," Nisaba said.

For the first time, the stranger's smile faltered.

"No," he said, softly. "Not yet."

The old stranger bowed low to Nisaba. It was not the frightened bow the others had given her. Re'u had seen friends part at the canal road with less sorrow.

"You smiled when you saw me," Nisaba said.

"Now you bow as people bow when a friend is leaving. Why?" Re'u added.

"You will see."

He found the western path with his stick and took his leave.

### 16.2.4. A count with a purpose

By midday the wool workers had begun. The sheep were gathered in smaller pens, and the loose spring wool came away beneath practiced hands. Nisaba watched the work openly. The first fear did not disappear, but work pressed around it. There were animals to hold, fiber to gather, cords to tie, water to carry, and children to keep from teaching themselves dangerous uses for a bronze visitor.

Abum-ilum wanted Nisaba taken to the storehouse. Two men tried to guide her there by standing on either side and explaining the importance of records. She walked instead to the shade beside Erishtum's wool baskets.

"She has chosen the wool," someone whispered.

"She has chosen shade," Erishtum said.

Nisaba watched Erishtum tie ten small bundles into a larger one.

"One," Nisaba said.

"Ten," said Erishtum.

Nisaba touched the cord around the whole bundle.

Erishtum understood. "One bundle. Ten smaller bundles. More wool than either of us wants to carry."

Nisaba looked at the work yard, where people had begun using her name with increasing confidence and no increase in agreement about what she was.

"Purpose," she said.

Erishtum pulled the knot tight. "Usually."

Many days later, at sunset, Ili-iddinam brought a fresh tablet. He had copied the flock total again. Nisaba set one stone on the tablet's bench and made a new mark beside it. She added another stone and another mark. Ili-iddinam watched her continue until nine stones stood in a row:

1 2 3 4 5 6 7 8 9.

He could follow the counts, although the signs were hers.

At the tenth stone, she gathered all ten into a bowl and wrote

10.

Ili-iddinam tapped the first mark. "You have returned to one."

“One group of ten,” Nisaba said.

He tapped the second.

She pointed to the empty bench beside the bowl. “No loose stones.”

Nisaba drew ten bowls in the dust, with ten small strokes inside each. Ili-iddinam followed her finger along them. “A hundred.”

She drew a larger enclosure around the ten bowls, then nine more enclosures like it. “Ten hundreds,” he said. “A thousand. We have names for those.”

“Then read these,” she said, writing 100 and 1,000 beside the drawings.

He pointed to the first numeral. “One hundred, no tens, no loose ones.” He followed the places in the second. “One thousand, no hundreds, no tens, no ones.”

Re’u brought the stones from the six circles. They regrouped them into three collections of a hundred and six mounds of ten. No stone was added or taken away. Nisaba wrote 360 beneath the arrangement.

“Three hundreds, six tens, no loose ones,” Ili-iddinam said.

“We change place after sixty,” Ili-iddinam said.

“These signs change after ten.”

He turned his tablet toward her and pressed the marks he used for six. “Six jars,” he said.

On a second scrap of clay he pressed the same marks. “Six sixties of sheep.”

Nisaba moved the scraps until they lay side by side. “The marks are the same.”

“The account tells us what place they hold. We leave an empty place when the account supplies what is missing.”

Re’u put the stones back into their six circles of sixty. Nisaba pointed first to the six large stones, then to the circles of sixty they named. Ili-iddinam’s marks gave the total through their place and through the account that held them. She wrote both reckonings together:

$$(6, 0)_{60} = 360_{10} = 3 \times 100 + 6 \times 10 + 0.$$

“Six groups of sixty and no loose ones,” she said, touching the two places on the left. The comma kept them apart. The small sixty below named their grouping rule; the small ten below the other numeral named hers. The two short lines between the numerals said that both gave the same count.

The young writer frowned at the zero after the six.

“We would leave that place empty.”

“How do you distinguish six from six sixties?” Nisaba asked.

“The account tells us.”

Nisaba touched the zero. “Now the numeral does.”

The old stranger laughed once from the darkness beyond the awning at the scribe’s apparent confusion.

“You asked her what she made one,” Ili-iddinam called to him.

The stranger turned toward the young man’s voice. “Did I?”

Ili-iddinam laughed, then stopped when no one joined him.

That evening Re’u found him copying Nisaba’s nine signs onto a scrap of clay small enough to close in his hand.

“Will you take them to Nippur?” Re’u asked.

Ili-iddinam covered the scrap with his palm, then appeared annoyed that he had done so. “When I can explain them without beginning with a bronze girl beneath a sheep awning.”

“Why leave out the best part?”

“Because I want them to listen after it.”

He uncovered the tablet and corrected the tail of the mark for seven.

## 16.3. Act II: The Count Becomes an Account

### 16.3.1. What changed

Nisaba remained after the spring reckoning. She remained through the wool gathering and the days when the cleaned fleeces were tied into bundles for storage. By then, people no longer stopped work whenever bronze feet crossed the yard.

Abum-ilum sent twice to Nippur for instructions and received two answers that did not agree. Ili-iddinam wrote both questions; Abum-ilum impressed the storehouse seal into the clay before they left Reed Gate. One ordered him to guard a manifestation belonging to the temple. The other ordered him not to make claims about a divine presence without witnesses competent to prove the claim. He solved the contradiction by declaring that Nisaba was under his protection whenever she happened to be near the storehouse.

Nisaba solved it by going where she pleased.

The temple sent three officials with a wooden carrying frame and six men to bear it before the wool gathering ended. They wrapped the crosspieces in cloth so the bronze would not be scratched, placed the frame beside Nisaba, and asked her to sit. She did. Six men took the poles. The frame did not rise.

They brought six more men and two ropes. The ropes tightened, the poles bent, and one bearer made a noise he later denied making. Nisaba watched them until Abum-ilum ordered everyone to stop before the temple lost either a manifestation or twelve useful men.

Then Amat-Sin called to Nisaba from the wool shelter. Nisaba stood, stepped off the frame, and crossed the yard. No one tried the ropes again.

One bent pole later supported the wool awning. For the rest of the season, anyone who told the story beneath it pointed to a different part of the bow. The temple bearers said the wood had been green. Ibiya said he had heard Nisaba's feet take root in the yard. Ahu-waqar said only that his sister had called and Nisaba had stood up, which no adult found sufficiently explanatory.

She walked with Re'u when the flock moved beyond the fields. She never appeared tired and never drank from his waterskin, but she stopped when the animals stopped and learned the ground by crossing it. At first she knew every count and almost nothing in the count.

At an irrigation cut, Re'u laid down a narrow plank before remembering how much bronze had defeated the carrying frame. Nisaba crossed it. The plank gave one small creak, less than it gave beneath Re'u. He looked back at her and then at the water.

"Do not ask me," she said.

"I was going to ask the plank."

She could say that one ewe lagged twelve paces behind another. Re'u could say she would be lame by evening if they did not remove the thorn between her toes.

Nisaba could divide the flock into two equal collections. Re'u could say which animals must remain together if he wanted the two collections to stay where he put them.

She could preserve the exact number of mouthfuls an animal took from a patch of grass. She could not tell from that number whether the animal had eaten enough.

"What do you see?" she asked him after a thin ewe refused water.

“Her sides. Her mouth. The way she stands.”

“Which one?”

“All of it.”

Whenever Re’u answered with the whole of what he saw, Nisaba asked him to name each sign separately. Re’u disliked losing sheep while she waited for every sign to have a name.

They kept working.

The work changed the way people approached her. At first they had brought offerings: a handful of grain, a twist of clean wool, a cup of beer placed where she stood. Nisaba examined each gift and left it untouched. Soon people brought disputes instead. The potter Ibiya and his brother agreed about how many jars had left their kiln but disagreed about where two had broken and who must bear the loss. Nisaba arranged pebbles until their accounts contradicted one another, then stopped.

“Which number is true?” one brother asked.

“The jar counts are the same,” she said. “You disagree about the two that broke.”

She would not decide what no count had established. By sunset, half of Reed Gate had heard that the bronze Nisaba could expose a false account. The other half heard that she refused justice.

Re’u understood both reports. People did not come to a goddess to hear that their own story was the part she could not count.

Children were less disappointed. They brought her questions that adults had learned not to ask aloud. Could she bend? Could she sleep? Was she hollow? Would she turn green if left in the rain? Nisaba answered the first two by crouching and by remaining upright through the night. For the other questions, she said, “I do not know.”

That answer frightened adults more than children.

The old stranger came and went by the canal paths. During the wool gathering he thanked Erishtum for a loaf she had not yet brought him. He told her the scorched edge had been better than the middle. When she baked the loaf two days later, one edge blackened against the oven wall. She carried it to him without cutting it. He tore off the scorched piece first.

A week later he asked Erishtum her name. He had known where the storehouse wall ended before Re’u led him there; now he sometimes stopped at a familiar turning and waited for someone to name the road. He remembered what had not happened and forgot what had.

Ibiya's daughter Simat-Adad called him moon-struck. Her mother told her not to say such a thing where an old man could hear, then used the same word beyond the wall that evening. After that, Reed Gate had a name for him.

People who called him moon-struck in daylight found him beside the wall after sunset and asked what he remembered.

"Will the rains come?" Lamassatum asked while Nisaba listened.

"I do not remember," he said.

"You claim to remember the days ahead."

"Some of them."

"What use is that?"

"Very little."

Lamassatum left him half a loaf anyway.

When people later credited Simat-Adad with naming the stranger, she denied it. She said everyone had already been thinking the word and she had merely been the first person careless enough to say it beside her mother.

When the stranger joined Nisaba beside the wall, he struck the ground once with his stick before sitting one forearm's length to her left. He did it each time. Nisaba never invited him and never moved away.

"Why do you remember only some days?" Nisaba asked.

The stranger turned toward her voice. "Why do you?"

Nisaba had no answer.

Before the feast in the longest days, a final tablet arrived from Nippur. Abum-ilum read the ruling aloud before the storehouse: Nisaba would remain wherever she chose to remain and would not be moved except by her own going. The temple had taken most of a season to command exactly what had already happened. Abum-ilum placed the ruling among the storehouse tablets with care. It was the first instruction on the matter that he could obey.

During a feast in the longest days, Amat-Sin and Simat-Adad brought Nisaba a circlet of green rushes threaded with the few flowers they had found beside the canal.

"Lower your head," the older girl told her.

Adults had knelt before Nisaba. Officials had tried to direct her with orders and claims of protection. Neither had moved her so readily as two children who could not reach.

Nisaba bent. The girls settled the wreath above her brow.

“What does it mark?” she asked.

“Summer,” said Amat-Sin.

“Nothing,” said Simat-Adad. “We made it for you.”

Nisaba straightened with the wreath still on her head. By midday, the small flowers had begun to curl against the hot bronze.

When the flowers dried, Amat-Sin tied the rush circle to the bent carrying pole above the wool awning. Simat-Adad maintained that this had been her idea. The wreath remained there after both girls had forgotten to argue about it.

Ahu-waqar, who had begun carrying wool bundles but was still young enough to forget where he set them, placed his palm against Nisaba’s forearm. He pulled it back at once.

“Hot,” he said.

Nisaba had stood in full sun since morning. Re’u poured a little water across the bronze. It darkened, ran over her wrist, and struck the dust. She watched the wet path disappear.

The two girls each took one of Nisaba’s hands and led her beneath the awning. She went with them.

“The same body changes with the day,” she said.

“Everything does.”

“A count does not.”

“A count changes when the things do.”

“The stone does not.”

Re’u picked up one of the counting stones. “This stone does not become another number either. But in the morning it was a sheep, at midday it was a bundle of wool, and now it is only in my hand.”

Nisaba took the stone. “The stone was never the sheep.”

“No. But if I forget what I made it stand for, its number will not help me.”

Amat-Sin had been listening from beneath the awning. While they spoke, she collected the smoothest stone from the last completed circle and carried it away to finish a game.

Nisaba counted the stones again. “There are fewer.”

Re’u looked through the gate at the flock. “There are not.”

Amat-Sin returned when Erishtum called her name, the missing stone clenched in one dusty hand. She had not stolen a sheep or altered a number. She had broken the arrangement that allowed someone else to recover the count.

Nisaba accepted the stone from her. “The number remained. The record failed.”

“The stone was being a house,” Amat-Sin said.

Nisaba considered this and restored the stone to its circle.

The next day she asked him to count a flock without using stones. Re’u made notches on a reed. Then she asked him to do it without the reed. He divided the animals among three pens and remembered a subtotal for each. Then she asked him to count them without stones, reeds, pens, or memory.

“I cannot.”

“Then number requires a material thing.”

“Counting requires me to keep my place. Today I use stone. Ili-iddinam uses clay. You use marks no one else has seen. The number may not require them. I do.”

Nisaba looked toward the tablet writer beneath his awning. He could show her a missed ewe, an erased mark, a broken reed, and an account restored because two records had been kept apart.

So he did.

Later, when she drew her signs in dust, she began to ask who needed to read them after the wind passed. Ili-iddinam taught her which clay held a clean wedge and how wet a tablet could be before a line collapsed. Erishtum taught her that a cord tied around wool could preserve a group where a written sign would be useless to the person carrying it. Re’u taught her that an animal marked yesterday might still have crossed the wrong gate today.

Nisaba gave them a notation that could travel farther than a mound of stones.

Not everyone who sat beside her brought a dispute or a lesson. An old basket maker named Shubultum came in the evenings and spoke while splitting reeds. She told Nisaba the names of her husband and two sisters, all dead, and which pair of hands had taught her to bend a rim without breaking it.

“What account should I make?” Nisaba asked.

“None. I know their names.”

“Then why tell me?”

"I wanted someone else to know who they were."

The following evening, Shubultum found Nisaba waiting beside the wall.

By the hottest month, they had found a practice between them. Re'u named what he noticed. Nisaba made the distinctions visible. He corrected what she had put together or kept apart. She showed him when two explanations that sounded different came to the same end.

The old stranger sometimes walked with them. His stick searched the ground before him. On steep paths he took Re'u's shoulder without apology; on level ground he refused it. His hands still trembled, though less than they had at the spring reckoning. He no longer knew the path around the deep irrigation cut or which household kept a water jar outside its wall. Re'u blamed age for the lost knowledge and warmer weather for the steadier hands, then avoided placing the two explanations beside each other. The stranger could sit beside Nisaba for a long time without asking what she was.

One day he lowered himself on her right, near enough that his sleeve touched her arm. Nisaba shifted away.

"You sit on my left," she said.

"Do I?"

"Every time. One forearm away. You strike the ground once before you sit."

He moved to her left and measured the distance with his hand. "Here?"

Nisaba moved his hand the width of two fingers farther. "Here."

The stranger smiled, though not as he had at the first reckoning. "I will remember."

Re'u came to suspect that the old man had once known exactly how to be near her and was now learning it from her.

### 16.3.2. Four kinds of absence

The number of animals present did not remain 360. The spring tablet still recorded what had been entrusted.

Abum-ilum ordered twenty-four animals transferred to another flock after a dispute over pasture. Ili-iddinam made a tablet for the transfer. Re'u watched the twenty-four go and set their stones in a separate bag. They were absent from his flock, but they were not lost.

Twelve adult animals died before the rains. Re'u preserved what evidence he could and reported each death. Those animals were absent in another way.

One hundred fifty lambs were born. Thirty died before the end of the dangerous first weeks. The surviving lambs were present, although none had existed when Abum-ilum made the spring tablet.

At the fold, Ili-iddinam tried to make the several tablets agree:

$$360 - 24 - 12 + 150 - 30.$$

Nisaba watched his stylus stop.

“It should be more than 360,” Re’u said.

“Why?” asked Nisaba.

“We added more than we took away.”

“How much more?”

Re’u moved stones while Ili-iddinam computed in clay.

The removals were

$$24 + 12 + 30 = 66.$$

The births added 150. The net change was therefore

$$150 - 66 = 84.$$

So the new total was

$$360 + 84 = 444.$$

Nisaba wrote the account a different way:

$$360 + (-24) + (-12) + 150 + (-30) = 444.$$

“You have made the dead less than nothing,” Ili-iddinam said.

“No,” said Nisaba. “I made the change negative.”

Re’u looked at the expression. The dead animals had not become negative sheep. The transfer had not made negative sheep either. The signs showed which way the count had moved: into the flock or out of it.

“The same sign for death and transfer,” he said.

Nisaba waited.

“The total needs that,” Re’u continued. “The account does not.”

He laid the twelve death stones apart from the twenty-four transfer stones.

Nisaba studied the two groups. Both reduced the total by the same operation. They did not mean the same thing.

“The total is right,” she said. “It does not say why they are gone.”

The old stranger, sitting against the wall, lifted his head.

“What have you counted?” he asked.

“Changes to the flock.”

“And what did the total leave out?”

Nisaba looked at the death stones and transfer stones. “Why the animals were absent.”

“You are learning to be dangerous,” he said. “Someone may prefer the total without those two piles.”

Nisaba appeared to consider this praise.

Shubultum, splitting reeds beside them, tied the death stones into one corner of a worn cloth and the transfer stones into another.

“The dead have names,” she said. “The transferred have another shepherd.”

Nisaba watched how she tied the two corners.

### 16.3.3. The tablet and the household

Erishtum used the count of 444 for a different purpose.

More animals meant more mouths to feed before they meant more wool to sell. The surviving lambs would not contribute equally to the next wool gathering, and the ewes nursing them would not endure the lean months on the same ration as dry adults. She asked Ili-iddinam to place the adult animals and lambs on separate lines.

He objected that the official total was already correct.

“Correct for what?” she asked.

He indicated the tablet. “For the account.”

“The account of what?”

“The flock.”

Erishtum turned to Re'u. "Did you teach him that answer?"

"He arrived with it."

They divided the present animals into two broad kinds:

$$324 \text{ adults} + 120 \text{ lambs} = 444 \text{ animals}.$$

The adult count followed from the original flock:

$$360 - 24 - 12 = 324.$$

The lamb count followed from births and early deaths:

$$150 - 30 = 120.$$

The two subtotals recombined to the same total. Nothing in the arithmetic forced the separation. Erishtum's purpose did.

She then separated the adults again: 180 breeding and nursing ewes, and 144 other adults. Re'u corrected three classifications after examining teeth and udders. Nisaba revised the marks without protest.

"Your first number was wrong," Ili-iddinam said to her.

"I put three animals on the wrong line," Nisaba answered. "The calculation followed the line."

The young man glanced at his tablet. He did not seem comforted.

"That is worse," he said.

"Usually," said Erishtum.

### 16.3.4. The first shortage

The rains were poor.

Water came, but not enough and not in the expected rhythm. Pasture that usually held through the cool months thinned early. The canal fields received what the local officials could preserve for barley, and the flock ranged farther for less.

Re'u noticed the change before the storehouse did. The animals returned sooner from grazing. The ewes crowded the trough. The lambs nosed the ground after the adults had moved on.

He estimated how long the pasture would hold.

“Thirty days short,” he told Abum-ilum.

“Twenty,” said the overseer.

“Thirty.”

“The tablet from last year says twenty-two.”

“The sheep have not read it.”

Abum-ilum did not smile. He was responsible for the barley in the storehouse, the claims already sealed against it, and the consequences of opening jars too early. If he released too much, no calculation would put the grain back.

“Bring the account tomorrow,” he told Ili-iddinam. “Bring all of it.”

The young writer looked at the several tablets now required to describe one flock.

“Which one is all of it?”

No one answered before the rain began again, lightly and without conviction.

## 16.4. Act III: Enough for Whom?

### 16.4.1. Forty-one measures

By the middle of the lean season, the failed rains had become visible in every trough and storage jar. At sunrise Abum-ilum showed the witnesses that the clay sealing the storehouse door was unbroken. He opened it; Ili-iddinam recorded that he had done so.

The inner rooms were cooler than the yard. Plastered bins and sealed jars stood above the damp that could creep through mud brick. Reed tags and clay sealings connected each group of containers to a purpose that Ili-iddinam could find on a tablet. Abum-ilum knew the stores by handling them. Ili-iddinam knew what Reed Gate would later have to answer for in Nippur.

The storehouse contained forty-one *gur* of barley when they opened the account.

The local measures nested inside one another:

$$1\ \text{gur} = 5\ \text{barig} = 30\ \text{ban} = 300\ \text{sila}.$$

The relationships were familiar to everyone present except Nisaba. Ili-iddinam explained them with the pride of a man briefly permitted to teach a goddess something.

"One *ban* is ten *sila*," he said. "Six *ban* make one *barig*. Five *barig* make one *gur*."

Nisaba wrote:

$$1 \text{ gur} = 5 \times 6 \times 10 \text{ sila} = 300 \text{ sila}.$$

"Yes," he said. "Except no one would write it like that."

"You understood it."

"Yes."

"Then one person would."

Ili-iddinam began to object, reconsidered the available evidence, and stopped.

The forty-one *gur* did not form one unclaimed heap. Thirty had been assigned to the flock. Six stood against household claims already witnessed. Five remained under the overseer's control for a failure elsewhere: a broken delivery, a sick work team, a field whose yield did not arrive.

For each amount, Abum-ilum indicated the containers and Ili-iddinam read the corresponding line. Neither man's memory was sufficient for the other man's responsibility.

Nisaba converted each amount to the smallest measure they were using:

$$30 \text{ gur} = 9{,}000 \text{ sila},$$

$$6 \text{ gur} = 1{,}800 \text{ sila},$$

$$5 \text{ gur} = 1{,}500 \text{ sila}.$$

Together:

$$9{,}000 + 1{,}800 + 1{,}500 = 12{,}300 \text{ sila} = 41 \text{ gur}.$$

The total balanced. No barley appeared.

### 16.4.2. The overseer’s plan

Abum-ilum began with the spring tablet.

“Three hundred sixty entrusted animals,” he said. “Thirty *gur* for thirty days.”

Ili-iddinam divided 9,000 *sila* by 360 animals and then by 30 days. Nisaba wrote the same relation in her notation:

$$\frac{9{,}000\ \text{sila}}{360\ \text{animals} \times 30\ \text{days}} = \frac{5}{6}\,\frac{\text{sila}}{\text{animal} \cdot \text{day}}.$$

“Five-sixths of a *sila* for each animal each day,” the young writer said.

Abum-ilum nodded. The plan used the barley already set aside, the count on the spring tablet, and the thirty days they had chosen. Another overseer could follow the marks and obtain the same result.

Re’u looked at the lines dividing one kind of animal from another.

“Which 360 animals?”

“The entrusted count.”

“Twenty-four of those are with another flock. Twelve are dead.”

“Their changes are recorded.”

“And 120 lambs are standing in the fold.”

Abum-ilum pressed his lips together. “The reserve was assigned against the entrusted flock.”

“The barley will be eaten by mouths, not by the tablet.”

The overseer’s irritation was not stupidity. If a new count could silently enlarge every obligation, then no reserve was a reserve. A shepherd could arrive with borrowed animals, point to their mouths, and consume grain owed elsewhere. If Abum-ilum ignored sealed claims whenever suffering appeared at his door, the storehouse would soon reward whoever could assemble the most visible suffering.

“I opened a jar once for a man who brought hungry animals to this gate,” Abum-ilum said. “Half belonged to his wife’s brother. When Nippur asked where the barley went, it did not ask how the animals had looked. It asked whose seal had been broken.”

“Give me a rule I can defend when you are gone,” he said.

Re’u had no answer yet.

### 16.4.3. Equal shares

Nisaba proposed the simplest revision. Use the 9,000 *sila* assigned to the flock, but divide it among all 444 animals for thirty days.

Before Ili-iddinam divided, Re'u compared the amounts. Across thirty days, 444 animals would require 13,320 daily shares. Nine thousand was less than that, so the share must be less than one *sila*. Half of 13,320 was 6,660, so the share must be more than half a *sila*.

"Between one-half and one," he said.

Ili-iddinam nodded and began the exact division.

$$\frac{9{,}000}{444 \times 30} = \frac{9{,}000}{13{,}320} = \frac{25}{37}.$$

The exact daily share was

$$\frac{25}{37}\,\frac{\text{sila}}{\text{animal} \cdot \text{day}}.$$

Ili-iddinam wanted a more usable number. Nisaba extended the division:

$$\frac{25}{37} = 0.675675675\ldots$$

"It does not end," he said.

"The quotient is exact," said Nisaba. "Its decimal repeats."

They could round the rate to 0.68 *sila* per animal per day. But multiplying the rounded rate back across the whole plan gave

$$0.68 \times 444 \times 30 = 9{,}057.6 \text{ sila}.$$

The rounded plan called for 57.6 *sila* beyond the available reserve.

Rounding down to 0.67 produced

$$0.67 \times 444 \times 30 = 8{,}924.4 \text{ sila},$$

which left 75.6 *sila* undistributed.

"Then do not use the rounded number to pour the barley," Re'u said.

Nisaba looked at him.

“Use it to speak,” he continued. “Keep the fraction to divide.”

She nodded.

The fraction preserved the exact division. The decimal made its size easier to say aloud.

“Then this is the equal plan,” Abum-ilum said.

“Yes,” said Nisaba.

“And it is correct?”

She looked once more at the division. “Yes.”

Erishtum rejected the plan for another reason.

“Equal measures for unequal animals,” she said.

She pointed to a nursing ewe, a dry wether, and a lamb.

“Those are each one animal. They are not each one need.”

She brought them into the pens. Re’u showed Nisaba the ewe whose lamb pulled milk from her twice before another lamb had risen, the old wether that had left part of its last measure, and a small lamb that could not reach the bottom of the vessel used for adults. Nothing in the division distinguished them.

When they returned, people were waiting for the goddess to defend her answer.

Nisaba drew a line through the proposed rate.

“The division is correct,” she said. “I made every animal one equal need. That was wrong.”

The admission traveled through the room differently from a judgment. Abum-ilum looked less reassured by her correction than he had by her confidence. Re’u trusted her more.

### 16.4.4. Feeding units

Re’u did not know the exact amount of barley each animal would require. No one did. The animals would still graze what they could find, weather would change their need, and sickness would make any fixed rate less trustworthy.

He did know enough to reject equality by headcount. After several days of observation, he proposed a weight for each kind of animal when they divided this barley:

- each of 180 breeding or nursing ewes counted as 1 feeding unit;
- each of 144 other adults counted as $\frac{3}{4}$ of a feeding unit; and
- each of 120 lambs counted as $\frac{1}{2}$ of a feeding unit.

The weights did not claim that a ewe ate exactly one *sila* of total food every day. They showed how much of this barley each kind should receive compared with the others, for this plan only.

Erishtum divided a small measure of barley into four equal-looking heaps. All four stood for the ewe's share, three for another adult's, and two for a lamb's. The comparison held whether she used a small measure or a large one.

Re'u asked her to fill a one-*sila* vessel. "This much for each breeding ewe each day," he said. "With whatever grazing remains. The others receive their smaller shares."

Erishtum looked through the doorway toward the pens. "We will have to change it if the pasture worsens."

"Or if they leave it uneaten."

She set the vessel beside the four heaps. The heaps showed the relative shares. The filled vessel showed how large Re'u proposed to make the ewe's daily share. It was a supplement chosen from what they had observed, not a measurement of everything the animal needed.

Nisaba calculated the daily total:

$$180 \times 1 = 180,$$

$$144 \times \frac{3}{4} = 108,$$

$$120 \times \frac{1}{2} = 60.$$

Therefore,

$$180 + 108 + 60 = 348 \text{ feeding units.}$$

For thirty days:

$$348 \text{ feeding units} \times 30 \text{ days} = 10{,}440 \text{ feeding-unit-days.}$$

At Re’u’s proposed rate of one *sila* per feeding unit per day, the plan required 10,440 *sila*:

$$10{,}440 \text{ sila} = 34 \text{ gur} + 4 \text{ barig}.$$

Only 30 *gur*, or 9,000 *sila*, had been set aside for the flock. The shortfall was

$$10{,}440 - 9{,}000 = 1{,}440 \text{ sila}$$

or

$$4 \text{ gur} + 4 \text{ barig}.$$

The weights made Re’u’s judgment about relative need visible. The weighted calculation also showed that 30 *gur* would not meet that plan.

Abum-ilum folded his arms. “Your better numbers have made the barley smaller.”

“No,” Nisaba said.

“I know.”

For the first time, the overseer sounded tired rather than angry.

#### 16.4.5. One household

The six *gur* marked for households stood in sealed jars along the north wall.

Two *gur* were associated with Re’u’s extended household. When Abum-ilum asked what Re’u would pledge for access to the contingency barley, Re’u looked toward those jars.

“Our two,” he said.

Erishtum turned so slowly that he wished she had moved faster.

“Our?”

“The household’s.”

“Which household?”

“Ours.”

“What did you make one?”

The old stranger lowered his head. Nisaba looked from Erishtum to Re'u with sudden attention.

"And what have you counted?" the stranger asked.

"Two *gur*," Re'u said.

"Barley is not a claim."

Erishtum answered him. "Then let us count the claims."

Erishtum brought out the small tablet witnessed after her husband's death. The tablet did not divide every jar and basket as though the people beneath one roof lived apart. Neither did it place every jar under Re'u's hand. Part of the claim maintained Erishtum and the children. The goods she had brought into the marriage, and what was kept for the children, had not passed into Re'u's hands when his brother died.

"You may answer with your labor," she said. "You may answer with wool owed to you. You may not make generosity out of my children's grain."

Re'u felt every witness in the storehouse become interested in the floor.

"I meant to preserve the flock that preserves us."

"Then say that. Do not say the cost is yours when you have put it in my jar."

Nisaba took the reed stylus from Ili-iddinam and drew a line through the single household entry. Below it she made separate marks for claims that had previously appeared as one.

The total remained two *gur*. The account became larger.

"The sum still balances because we left out her claim," Nisaba said. "That did not remove it."

Erishtum inspected the new lines. "No. Someone still has to pay it."

The witnesses did not leave the matter inside the storehouse. By evening, people in the work yard knew that Erishtum had stopped Re'u from pledging the household grain. Ibiya said she had shamed him before an overseer. Lamassatum said Re'u had first shamed himself by naming another person's grain. Before dark, Simat-Adad had heard a version in which Erishtum struck his hand away from an open jar. She repeated it only to explain why it could not possibly be true. None of these accounts described what had happened, but each found an audience prepared to improve it.

Re'u carried water to the wool shelter after dark. Erishtum was sorting fiber by lamplight. Ahu-waqar slept against the wall; Amat-Sin separated tangled locks with solemn inattention. The spindle, baskets, sealed jars, and tablet occupied different parts of the room, but they belonged to one working

arrangement. If the barley disappeared, the children would eat less. If the wool work stopped, an exchange the household expected would disappear later. If Re’u lost the flock, a different stream of wool would disappear after that. The jars did not all belong to one person in the same way, though they stood against the same wall.

“I was trying to keep all of it,” he said.

Erishtum did not look up. “You were trying to keep what you see from the field.”

“I see this room.”

“When you enter it.”

He set down the water. Amat-Sin discovered a knot and began working at it with both hands.

“If the flock goes,” Re’u said, “my contract goes with it.”

“I know.”

“Then you know why I offered the grain.”

“You looked at my jars before you named your wool.”

That was not altogether fair. It was fair enough that Re’u could not answer quickly.

Erishtum finally set aside the wool. “I knew the stores while my husband was alive,” she said. “I knew the wool. I knew which obligation belonged to which jar. When he died, men began asking me to prove what I already knew.”

Re’u sat against the wall. “What should the account say?”

“It should not say every jar is mine and none of it serves the house. That would lie in the other direction. It should say which grain keeps the children, which came with my marriage, which depends on your flock, and which you may actually promise. Then we can see what saving the flock costs each of us.”

“Ilum-bani would have offered the jars,” Re’u said.

“Ilum-bani offered anything when other people were watching.”

“He would have kept the flock.”

“He would have come home after dark and asked whether I had found a way to do it.”

Re’u remembered his brother making promises in the yard and Erishtum making them possible indoors. He had mistaken the order before.

“That will be a larger tablet.”

“Then use more clay. Write which grain you mean.”

In the doorway, Nisaba repeated the sentence. She had arrived without either of them hearing her.

“Sometimes,” Erishtum amended.

Nisaba entered and examined the work. She lifted the spindle whorl, turned it once, and returned it precisely where it had lain.

“This makes the thread?” she asked.

“Hands make the thread,” Erishtum said. “The spindle helps the hands keep twist. The whorl helps the spindle keep turning. The wool matters. So does practice. If you write that the whorl made the thread, the tablet will be shorter and wrong.”

Nisaba looked at Re’u.

“Do not ask me,” he said. “I have already made one household.”

Erishtum laughed despite herself. The sleeping child stirred, and all three adults became quiet.

The following morning Ili-iddinam made a new account. He did not divide every object among named owners. He recorded the claims that mattered to the pledge they were discussing and left the rest of the household intact. The tablet could not make a shepherd bound by contract, an overseer, a widow, and a young writer equal. It could make it harder for one of them to conceal a transfer inside a total.

Nisaba read the new lines in the open yard where everyone waiting at the storehouse could hear. Her presence did not settle whether Erishtum had behaved properly. It placed the changed account before witnesses. For the rest of the season, anyone who called the two *gur* simply Re’u’s grain had to do so against a visible record.

### 16.4.6. The fourth plan

Re’u walked through the adult pen before dawn.

Some animals would not carry the flock into another season. Twenty-four of the other adults were old, chronically lame, or unlikely to survive the longer range ahead. Keeping them within the feeding plan reduced what could go to the breeding ewes and young stock. Removing them would mean sale where a buyer could be found and slaughter where one could not.

There was no notation that made the choice clean.

After the twenty-four were removed, the planned flock contained:

- 180 breeding or nursing ewes;
- 120 other adults; and
- 120 lambs.

That was 420 animals:

$$180 + 120 + 120 = 420.$$

The weighted daily requirement became

$$180 \times 1 + 120 \times \frac{3}{4} + 120 \times \frac{1}{2} = 180 + 90 + 60 = 330 \text{ feeding units.}$$

For thirty days:

$$330 \text{ feeding units} \times 30 \text{ days} = 9{,}900 \text{ feeding-unit-days.}$$

At one *sila* per feeding unit per day, the plan required exactly

$$9{,}900 \text{ sila} = 33 \text{ gur.}$$

The jars already set aside for the flock supplied 30 *gur*. The remaining need was 3 *gur*.

Abum-ilum could release that amount from the 5-*gur* contingency and retain 2 *gur* for other failures. He would do it only against a witnessed obligation and a minimum count at the spring reckoning.

Re’u offered his next wool allotment and, if that did not cover the advance, his labor. If he failed to return the agreed minimum flock, he would lose the contract.

Erishtum made Ili-iddinam read each kind of animal and each promised share again. Her claim and the children’s claim did not appear as security.

“Now it is your sacrifice,” she told Re’u.

She did not say it kindly. She did stand beside him when he made his mark.

## 16.5. Act IV: Re’u’s Mark

### 16.5.1. What arithmetic could settle

Two days after Re’u completed the fourth plan, the witnesses gathered in the storehouse at midday.

Ili-iddinam had prepared four tablets. The first preserved the original entrustment of 360 animals. The second recorded births, deaths, and transfers leading to the present count of 444. The third divided the animals by age and feeding claim. The fourth described the proposed division of barley, removal of twenty-four adults, release of 3 contingency *gur*, and Re’u’s obligation.

Abum-ilum had asked Ili-iddinam to call the release necessary.

“Who will witness that it was?” the young man had asked.

The overseer studied him long enough for the room to become quiet. “Write proposed.”

Ili-iddinam wrote proposed.

Abum-ilum asked Nisaba to confirm the final tablet.

She read each line. She recomputed the totals. Then she set the tablet down.

“Is it correct?” he asked.

“If these are the animals, if these weights describe the intended shares, if the plan lasts thirty days, and if one feeding unit receives one *sila* each day, then 9,900 *sila* follow.”

“So it is correct.”

“Only if each of those things is true.”

“Which plan should I allow?”

Nisaba looked at the four tablets.

“The numbers do not choose.”

The overseer stared at her. For months, people had brought Nisaba questions because her answers did not tire, flatter, or forget. Some had begun calling any answer she gave a judgment. Now, when judgment was wanted, she had returned the problem.

“Then what good are you?” Abum-ilum asked.

Several witnesses inhaled at once.

Nisaba did not appear offended.

“I can show what follows from what you have made one.” Nisaba turned the four tablets toward Re’u. The witnesses turned with them.

Re’u picked up the tablet that described the fourth plan. “Come to the pens.”

### 16.5.2. What Re’u knew

Re’u walked the witnesses through the pens.

He did not describe the twenty-four marked adults as useless. They had produced wool, sired lambs, endured dry seasons, and taught younger animals where to find water. Their history did not alter the barley available to them now. Their removal would give the remaining flock a better chance. It would also make their deaths or sale part of Re’u’s decision rather than something he could later blame on weather.

He showed Abum-ilum the breeding ewes whose loss would narrow the next season before it began. He showed him lambs strong enough to survive if the ration lasted. He showed him two animals he expected to lose under every plan.

“Then why feed them?” the overseer asked.

“I expect to lose them. They are still alive.”

“You told me to remove twenty-four because you expect them not to survive.”

“Yes.”

“What is the difference?”

Re’u looked from one animal to the next. He could state differences in age, teeth, gait, appetite, and breeding value. No mark told him where knowledge ended and choice began.

“I cannot draw a clean line,” he said. “I think removing those twenty-four gives the others a better chance. I may be wrong about which ones to keep.”

Abum-ilum looked again at the two animals. Re’u had no further answer for him.

Erishtum described the household claims. She named what would happen if the sealed barley was treated as spare: who would eat less first, which wool work would stop, and which debt would remain after the grain disappeared.

She did not ask them to spare the household merely because it was a household. She named the people, jars, work, and debts that Re'u's word *ours* had pressed together.

Ili-iddinam described the record. He could preserve the original total without pretending it was the present count. He could preserve the present count without pretending every present animal carried the same obligation. He could place Re'u's pledge on the same tablet as the contingency release and keep Erishtum's claim outside it.

Abum-ilum described what he would have to answer for. Two contingency *gur* would remain. If another delivery failed, people would ask why he had released the other three. The four tablets could defend the reasoning, but they could not replenish the jars.

Each person knew something the others needed.

### 16.5.3. The choice

Re'u chose the weighted plan.

Before sunrise he carried twenty-four short red cords into the adult pen. Under the fourth plan, the barley allocated to the flock could provide the planned shares for 420 animals. Re'u had chosen that twenty-four would leave it. The arithmetic had not named them.

He began with the obvious ones: a wether that placed no weight on one foreleg, two animals whose teeth could no longer break the coarse feed, another with a swelling that had returned after every treatment. He tied one cord loosely around each neck.

The fifth animal was an old wether with a split left ear. For years it had stayed near water when the rest of the flock wandered toward better grass. Younger animals had sometimes followed it back when heat made the longer route dangerous. It stood soundly enough now. It would probably lose condition first when the range lengthened.

Re'u held the cord for some time before tying it. The wether lowered its head to nose his wrist, searching for salt. Its history was not another mouth, another feeding unit, or another line on the tablet. Its history also did not fill a jar.

Nisaba followed without advising him. After the twelfth cord, she counted those marked and those not marked. After the twenty-fourth, she counted again.

"Twenty-four," she said.

The stranger stood outside the hurdle, his stick resting lightly against one leg. “What have you counted?”

“Animals marked to leave the flock,” Nisaba answered.

“And what has he chosen?”

She looked toward Re’u.

“That is not another count,” he said.

The trader accepted thirteen animals. They went toward the city under another man’s staff. The remaining eleven were killed near the work yard, where the meat could be divided promptly and the hides cleaned before the day warmed.

Re’u gave the killing to no one else. He held each animal, kept the blade sharp, and remained through the cleaning. Erishtum directed the division of meat among the households whose labor would preserve it. Shubultum brought broad reed trays. Lamassatum measured salt from a jar she had hoped not to open until winter. Simat-Adad carried water after protesting that she had never called anyone moon-struck, and Ahu-waqar guarded the red cords with such attention that he forgot the knife Erishtum had sent him to fetch. Ili-iddinam recorded eleven slaughtered and thirteen transferred by sale. The two lines produced the same subtraction from the feeding count. They did not describe the same event.

When they reached the split-eared wether, Re’u almost sent it with the trader instead. That would have changed where the death occurred, not necessarily whether it occurred. He kept the animal back and did the work he had chosen.

By midday the red cords lay in a bowl. Nisaba touched one and then withdrew her hand. Re’u could not tell whether the bronze retained anything of what it touched. He knew that he would.

Abum-ilum released 3 *gur* from the contingency reserve. The barley available to the flock became

$$30\ \text{gur} + 3\ \text{gur} = 33\ \text{gur} = 9{,}900\ \text{sila}.$$

Two contingency *gur* remained. The 6 *gur* under household claims remained sealed.

Ili-iddinam read Re’u’s obligation aloud. Re’u pledged his next wool allotment and, if the value credited to it did not satisfy the advance, the labor required to complete the obligation. If he failed to return at least six sixties of animals at the next spring reckoning, Abum-ilum could place the flock with another shepherd.

“Six sixties,” Re’u said. “Not a sign whose size must be guessed.”

Ili-iddinam added words making the magnitude explicit.

Re’u pressed his mark into the clay.

His mark did not make him Abum-ilum’s equal. It preserved what each man had undertaken before witnesses who might outlast either man’s wish to remember it differently.

Erishtum inspected the line naming the security. Her claim did not appear there.

Abum-ilum sealed the tablet.

Nisaba made no declaration. She watched Re’u’s thumb leave the clay.

### 16.5.4. Thirty days

The plan did not unfold like its table.

On the fourth day, the weather warmed and the animals found more grazing than Re’u expected. On the seventh, a ewe became sick and consumed less. On the twelfth, wind spoiled part of one open measure before anyone covered it. On the seventeenth, two lambs entered the wrong pen and were nearly counted twice.

The daily total was what they meant to pour, not what every day allowed. Re’u and Erishtum adjusted the shares promised to each kind of animal while Ili-iddinam marked every change large enough to empty the jars too soon.

For the first seven days, Nisaba kept another account beside the clay one, using her own signs:

| Day | Planned *sila* | Issued *sila* | Difference from plan |
|---|---|---|---|
| 1 | 330 | 330 | 0 |
| 2 | 330 | 326 | −4 |
| 3 | 330 | 328 | −2 |
| 4 | 330 | 318 | −12 |
| 5 | 330 | 324 | −6 |
| 6 | 330 | 329 | −1 |
| 7 | 330 | 327 | −3 |
| **Total** | **2,310** | **2,282** | −28 |

The last column compared what had been issued with what they had planned:

$$\text{difference} = \text{issued} - \text{planned}.$$

A negative result meant that less had been issued than planned. It did not mean that anyone had issued negative barley.

On the fourth day, the large difference came from improved grazing. If Ili-iddinam had simply replaced 330 with 318, no mark would remain of what they had meant to pour. By preserving both columns, he could show the first plan beside what they had actually poured.

“Which number governs tomorrow?” he asked.

“Neither,” Re’u said. “We have not reached tomorrow.”

Nisaba pointed to the remaining barley. “What remains changes what we can pour.”

“Yes.”

She pointed to the daily plan. “Which animals they are changes what should be offered.”

“Yes.”

“And the animals change what they will take.”

“Yes. That is why the table keeps changing.”

At the end of the week, 28 *sila* remained beyond the planned balance. That did not justify increasing the next day’s issue by exactly 28. The difference told them what remained; it did not tell them what to pour tomorrow. Those 28 *sila* could answer a later day when wind, sickness, or worse grazing moved use in the other direction.

The measures themselves imposed another limit. No one could pour a fraction with the exactness of a mark on clay. They used vessels, hands, and judgment; grain settled differently, spilled, and clung to the sides. The tablet could show a clean measure even when no jar had been poured so cleanly. Adding decimal places would not repair that mismatch.

When the lambs crossed pens on the seventeenth day, the count on the distribution tablet was two higher than the count made at the outer gate. Nisaba recomputed the additions and found no error. Re’u walked the fences. He found a loosened reed hurdle and two sets of fresh tracks.

“The arithmetic was correct,” Ili-iddinam said when they repaired the entry.

“The addition was right,” Re’u answered. “We counted two lambs twice.”

They added a second count at the gate. From then on, the person who divided the animals among pens did not also give the final number at the outer gate. A second count cost time. It cost less than feeding animals twice on clay and once in the yard.

Nisaba asked for a rule behind every adjustment.

“If this ewe does not eat her share,” she asked, “where does it go?”

“Today, to those two.”

“Why those?”

“They lost condition yesterday.”

“By what measure?”

Re’u showed her the ridge of a spine beneath wool, the hollow beside a tail, the resistance of an animal that still had strength to object.

Nisaba touched the same signs. Her bronze fingers could distinguish shape and distance. Re’u did not know whether she could feel warmth.

“Can this be written?” she asked.

“Some of it.”

“Can all of it be learned?”

“Perhaps. Not before they eat.”

She looked again at the animals. “Then they may have to eat before the whole of it can be put on clay.”

“They usually do.”

She gave no sign that she heard either pride or apology in that answer.

At the end of thirty days, the flock had consumed slightly less than the most they had planned to pour because grazing had briefly improved. That did not mean Abum-ilum had been wrong to open the contingency jars. Without them, Re’u could not have kept the promised shares long enough to respond to what happened. Barley left in a jar did not mean the jar should have remained sealed. The unissued balance remained assigned to the flock for the lean weeks that followed. It was part of the 3-*gur* advance and did not return to the contingency reserve.

### 16.5.5. The second spring

At the next spring reckoning, 384 animals passed through the gate.

Re’u counted six circles of sixty and then twenty-four more:

$$384 = 6 \times 60 + 24 = (6, 24)_{60}.$$

The total exceeded the six sixties required by the tablet by 24. It was also 60 below the 444 present before they divided the barley:

$$444 - 384 = 60.$$

Twenty-four of that reduction had been chosen before the barley was divided. The rest came through the season in deaths, losses, and changes no plan had prevented.

The count was enough for Re'u to retain the flock. It was not enough to erase the obligation. His wool allotment went first against the 3-*gur* advance, and the remaining balance would be met through labor.

Abum-ilum did not congratulate him. Re'u had not expected congratulations from a storehouse. The overseer did renew the arrangement.

Erishtum received the household barley under the claims the second account had preserved. Ahu-waqar and Amat-Sin ate from it without knowing how close their grain had come to being called Re'u's sacrifice. That ignorance was not ingratitude. Children should not have to study an account before they are permitted to eat.

Ili-iddinam copied the final count beneath the earlier entries. He did not press the history into one total. The tablets would preserve both counts: the animals first entrusted and the animals present later, along with each birth, death, and transfer, how the barley had been divided, and whose wool and labor stood against it.

Nisaba read it from beginning to end.

"Correct?" the young writer asked.

She looked at him.

"Another hand can follow it," she said.

He smiled. A year earlier he had objected when Erishtum asked what a correct total was correct for. Now he copied the words beside the final count.

"I will take the signs to Nippur," he said.

"You intended to do that when you hid them under your hand," Re'u said.

The young man had the grace to look embarrassed. "I intended to take the marks. Now I will take the account of where I learned them."

Nisaba looked at the new line naming Reed Gate. “The marks would work without my name.”

“In Nippur they will ask whose account they came from,” Ili-iddinam said.

Then he glanced toward the old stranger. “He said you would answer that.”

Nisaba turned. The stranger had tilted his head toward Ili-iddinam with the attentive stillness he used for an unfamiliar voice.

“Who is speaking?” he asked.

“Ili-iddinam,” the young writer said, more sharply than his name required.

“Yes,” said the stranger. “I will remember.”

“A prophet,” someone whispered behind Re’u.

“Not prophecy,” the stranger said. “Memory.”

No one appeared comforted by the correction.

### 16.6. Back to the Stones

After the reckoning, Re’u found Nisaba arranging the original counting stones in the yard. She had made six rows of sixty. Then she made ten rows of thirty-six, twelve rows of thirty, fifteen rows of twenty-four, and eighteen rows of twenty.

The arrangements looked different. Each contained 360 stones.

Multiplication described their rectangular structure:

$$360 = 6 \times 60 = 10 \times 36 = 12 \times 30 = 15 \times 24 = 18 \times 20.$$

“How many answers?” Re’u asked.

“One number,” Nisaba said. “Several arrangements.”

The stranger came along the wall, tapping until his stick found the first row. He lowered himself to the ground and ran his fingers over the stones. Nisaba returned the stones to six rows of sixty and drew a line after the third row. Two sets of rows, with three rows in each. Within every row, she marked six groups of ten, then separated those groups into two sets of three. Within each group of ten, she made two piles of five.

Re’u followed the divisions from the whole arrangement to one little pile: two sets of rows, three rows in each, two sets of groups in each row, three groups in each set, two piles in each group, five stones in each pile. All 360

stones remained. The smaller numbers counted the groups within groups; they were multiplied to recover the whole, not added as separate piles.

Nisaba spoke the group counts as she followed the lines:

$$360 = 2 \times 3 \times 2 \times 3 \times 2 \times 5.$$

Each count in that chain could make equal groups only as a single whole or as ones. Five stones could become a pile of two and a pile of three, but those piles would not be equal.

Then she brushed away the lines and made eighteen rows of twenty. She divided the rows into two sets of nine, then each set into three bundles of three rows. Within each row, she made two sets of ten, each containing two piles of five. The stones had different neighbors now. The group counts were two, three, three, two, two, and five.

"Again," Re'u said.

Nisaba tried twelve and thirty. The order changed. The final group counts did not.

Then she drew one of the old circles apart and worked with its sixty stones. She began with six groups of ten:

$$60 = 6 \times 10 = (2 \times 3)(2 \times 5) = 2 \times 2 \times 3 \times 5.$$

She began again with twelve groups of five:

$$60 = 12 \times 5 = (2 \times 2 \times 3) \times 5.$$

The path changed again. The last factors did not.

The stranger smiled toward the sound of the stones. "A road may turn at every hill and still be the road to one city."

"Must it always end there?" Nisaba asked.

"You have watched it happen twice," the stranger said. "Twice is not always."

Nisaba began another arrangement. She did not ask why he sounded certain that there was more to learn.

### 16.7. Coda: The Beginning

The old stranger left before sunrise on the day after the reckoning.

Re'u found him at the western path fastening a strap. The old man's hands trembled less than they had at the first spring reckoning, but the leather had split and would not hold.

"Who is there?" the stranger asked.

"Re'u."

The old man repeated the name carefully. He had leaned on Re'u's shoulder through half the year.

Re'u repaired the strap without asking where the old man intended to go.

Footsteps sounded behind them. Erishtum had brought half a loaf, scorched at one edge. Ili-iddinam and Abum-ilum followed, each perhaps unwilling to let the other become the only witness to the departure.

The old man turned from one voice to the next without recognition. Re'u gave him the introductions again. "Abum-ilum oversees the storehouse. Ili-iddinam writes its tablets. Erishtum is of my household and knows what the wool and jars contain better than I do."

The stranger repeated each name. At the first spring Re'u had seemed only to confirm what he knew. Now he listened like a traveler learning where he had stopped.

"What is this place called?" he asked.

"Reed Gate."

Nisaba had followed the stranger from the fold. She stopped one forearm's length to his left and struck the ground once with one bronze finger. The bronze of her face held no expression Re'u could name.

"Do you know me?" she asked.

He turned toward her without hesitation. "Yes."

"Where did you learn the question?" she asked.

For a long time, the old man said nothing.

"At the beginning."

"This is the beginning."

"Not mine."

Nisaba was silent after that answer. The stranger turned his face toward the sheepfold. Behind Nisaba, animals shifted against the hurdles, a stylus scratched beneath the awning, and the six circles of stone still lay in the yard. Above them, the rush wreath had dried almost white around the bend in the old carrying pole.

"Perhaps I have grown wiser," he said. "Or perhaps the world has simply grown simpler."

Re'u thought of the four tablets required to describe one flock, the barley still owed, the animals that had not returned, and the children eating grain that had nearly vanished inside the word *household*.

"Or perhaps it has not," he said.

The old man laughed. "No. Perhaps not."

He raised one hand toward Nisaba's bronze brow and stopped, uncertain. She took him by the wrist and completed the motion.

He turned toward the road. He walked until the reeds concealed him.

## 16.8. Arithmetic Review and Practice

The story has introduced the ideas first. This review now names the ideas, states the rules plainly, and gives you calculations to try.

### 16.8.1. First Reckoning: What the Count Contains

#### 16.8.1.1. Try before reading

Return for a moment to the stones in the yard. Re'u calls what he sees six circles; Ili-iddinam records 360 sheep. How can both descriptions be correct?

Then consider Nisaba's new sign in $(6, 0)_{60}$. How can a mark for no loose group preserve the place value of another mark? Compare your answers with the explanation below, then revise them where needed.

The stranger's lesson separates five related things:

- a sheep is a **thing**;
- the flock has a **quantity**, an amount that can be counted or compared;
- 360 is the **number** describing that quantity;
- the written marks 360 are a **numeral**, a representation of the number; and
- one sheep is the **unit** selected for this count.

The stone is neither the sheep nor the number. It helps Re'u preserve a **one-to-one correspondence**: one stone for each sheep, with none repeated and none omitted. The question "What did you make one?" identifies the unit and boundary before calculation begins.

In decimal notation, ten ones make one ten, ten tens make one hundred, and ten hundreds make one thousand:

$$10 \times 1 = 10, \qquad 10 \times 10 = 100, \qquad 10 \times 100 = 1{,}000.$$

Read the places from right to left: ones, tens, hundreds, thousands. Each place is worth ten times the place beside it on the right. Thus 360 records three hundreds, six tens, and no ones:

$$360 = 3 \times 100 + 6 \times 10 + 0 \times 1.$$

And 1,206 records one thousand, two hundreds, no tens, and six ones:

$$1{,}206 = 1 \times 1{,}000 + 2 \times 100 + 0 \times 10 + 6 \times 1.$$

The comma in 1,206 makes the long decimal numeral easier to read; removing it does not change the value. The comma inside our base-sixty parentheses has a different job: it separates whole sexagesimal places, each of which may need more than one decimal digit to write.

Zero may name an empty quantity, as in 0 sheep, or hold a place in a numeral. In 205, it records that there are no tens between two hundreds and five ones:

$$205 = 2 \times 100 + 0 \times 10 + 5.$$

The equals sign says that two expressions represent the same value; it does not mean merely that an answer follows. Thus

$$6 \times 60 = 360$$

is a claim that can be checked.

The signs $<$ and $>$ express order. For example, 444 animals $>$ 360 animals compares two counts with the same unit. The inequality does not explain why the count changed, whether the same animals are present, or whether the two counts use the same boundary. A comparison such as 12 days $>$ 8 animals does not make a meaningful claim about the quantities merely because $12 > 8$ as numbers.

The signs Nisaba writes from 1 through 9 and then combines in 10 belong to our modern decimal notation. Our usual notation is **base ten**: moving one place left multiplies its value by ten. Old Babylonian calculation often used **base sixty**. Each sexagesimal place can hold a value from 0 through 59 before another place is needed. This chapter uses a modern comma and subscript to expose the places:

$$(6, 0)_{60} = 6 \times 60 + 0 = 360_{10},$$

$$(7, 24)_{60} = 7 \times 60 + 24 = 444_{10}.$$

For these two-place numerals, the part before the comma counts sixties and the part after it counts loose ones. The small number below the numeral is a **subscript** naming its base. It tells you to use places organized by sixty or by ten; it is not another amount to add or multiply.

The numeral changes with the base; the number does not. Our explicit zero in $(6, 0)_{60}$ is a modern teaching device, not a reproduction of Old Babylonian notation. Sixty is useful because it divides evenly by 2, 3, 4, 5, and 6, among other numbers.

#### 16.8.1.2. Practice and transfer

Try these before checking the responses.

1. A tray contains one token for each of 24 completed orders. Identify the things, number, numeral, and unit.
2. Convert $(5, 12)_{60}$ to decimal notation.
3. Explain the different jobs performed by zero in "zero defects" and in 407.
4. A report says that two departments each completed "12." What must you know before concluding that their outputs were equal?
5. Explain why 12 days $>$ 8 animals is not a useful comparison of quantities, even though $12 > 8$.
6. Expand 2,040 into thousands, hundreds, tens, and ones. What does each zero preserve?

Check Your Reasoning

1. The completed orders are the things; twenty-four is the number; 24 is the numeral; one completed order is the unit.
2. $(5, 12)_{60} = 5 \times 60 + 12 = 312_{10}$.
3. "Zero defects" gives the count of defects. The zero in 407 is a placeholder showing that there are no tens between four hundreds and seven ones.
4. You need at least the unit and boundary: twelve what, completed under what definition, and during what interval?
5. The numbers can be ordered, but the measured quantities use incompatible units. A useful inequality must compare quantities made comparable by their units and definitions.
6. $2{,}040 = 2\times1{,}000+0\times100+4\times10+0\times1$. The first zero preserves the hundreds place and the second preserves the ones place. Omitting them would turn the numeral into 24, a different number.

### 16.8.2. Second Reckoning: What an Account Preserves

#### 16.8.2.1. Try before reading

Without performing the exact calculation, decide whether

$$360 - 24 - 12 + 150 - 30$$

should be greater than or less than 360. What in the expression supports your prediction?

Now imagine that every removal is recorded only as a negative change. What would the total preserve, and what would it erase about the animals and the decisions that removed them?

Arithmetic operations transform represented quantities according to rules. The rules matter, but an operation acquires its meaning from the situation and units around it.

#### 16.8.2.2. Addition and subtraction

Addition combines compatible quantities:

$$324 \text{ animals} + 120 \text{ animals} = 444 \text{ animals}.$$

Adding 324 animals to 120 days produces no meaningful total even though a calculator can add the numerals. Subtraction may find a remainder, compare two values, or reverse an addition. The expression $444 - 360 = 84$ gives a difference; it does not identify what caused it.

#### 16.8.2.3. Multiplication and division

Multiplication forms equal groups or scales a quantity:

$$144 \text{ animals} \times \frac{3}{4}\frac{\text{feeding unit}}{\text{animal}} = 108 \text{ feeding units}.$$

Division may ask how many groups of a selected size can be formed, as in $360 \div 60 = 6$, or how much belongs in each of a selected number of groups, as in $360 \div 6 = 60$. The same operation answers different questions only because the unit and grouping supply its meaning.

#### 16.8.2.4. Signed quantities describe direction

A positive or negative sign can describe a value relative to a chosen zero or the direction of a change.

$$(+150) + (-30) + (-24) + (-12) = +84.$$

The final count is 444, not 84; +84 is the change from the starting level of 360. In −24, the minus sign belongs to a negative value. In 360 − 24, it directs subtraction. Neither use creates negative sheep.

#### 16.8.2.5. Expressions, statements, malformed notation, and undefined operations

Arithmetic notation has grammar. That means errors have different kinds.

A well-formed **expression** specifies a calculation and, when it is defined, represents a value:

$$360 - 24 - 12 + 150 - 30.$$

A **statement** makes a claim:

$$360 - 24 - 12 + 150 - 30 = 444.$$

This statement is true.

The statement

$$360 - 24 - 12 + 150 - 30 = 445$$

is meaningful and false.

The string

$$360 + \times 24$$

is malformed under ordinary arithmetic grammar. It has not yet expressed a claim that can be true or false.

Some well-formed expressions are **undefined**. For example, $12 \div 0$ would require a number $x$ for which $0 \times x = 12$. No such number exists in ordinary arithmetic, so the expression is undefined.

Do not confuse an undefined operation with a result that falls outside a selected kind of number. The calculation $3 - 5 = -2$ is defined, although its result is not a nonnegative whole number. Likewise, $1 \div 2 = \frac{1}{2}$ is defined,

although its result is not an integer. Expanding the available kinds of number admits those results; it does not define $12 \div 0$ in the arithmetic used here.

#### 16.8.2.6. Grouping and order

Parentheses tell us what to treat as a unit of calculation.

$$360 - (24 + 12) = 324.$$

By contrast, $(360 - 24) + 12 = 348$ removes 24 and then adds 12. The grouping changes the claim.

Use the conventional order of operations:

1. evaluate grouped expressions;
2. evaluate exponents;
3. multiply and divide from left to right; and
4. add and subtract from left to right.

The familiar mnemonics are less reliable than understanding what the notation says. Multiplication and division share a level; addition and subtraction share a level. Neither division nor subtraction should be silently moved around as though it were commutative.

#### 16.8.2.7. Properties that permit rearrangement

Combining 24 transferred animals and 12 dead animals gives the same removal total whichever group we name first:

$$24 + 12 = 12 + 24 = 36.$$

Likewise, six rows of ten stones and ten rows of six stones contain the same number:

$$6 \times 10 = 10 \times 6 = 60.$$

Addition and multiplication are **commutative**: exchanging the order leaves the result unchanged. They are also **associative**: changing the grouping leaves it unchanged. For example, $(24 + 12) + 30 = 24 + (12 + 30) = 66$.

We can state these patterns generally with letters. Here $a$, $b$, and $c$ stand for numbers; they do not name another kind of operation. Writing $ab$ means $a \times b$. With those conventions, the same rules say:

$$a + b = b + a, \qquad ab = ba.$$

$$(a + b) + c = a + (b + c), \qquad (ab)c = a(bc).$$

The distributive property lets us split a multiplication into smaller parts. Seventeen groups of twelve are seventeen groups of ten together with seventeen groups of two:

$$17(10 + 2) = 17 \times 10 + 17 \times 2 = 170 + 34 = 204.$$

The parentheses collect a quantity, and writing a number beside them means multiplication. Using letters to state the same relationship:

$$a(b + c) = ab + ac.$$

Adding no animals to twelve leaves twelve: $12 + 0 = 12$. Multiplying twelve by one also leaves twelve: $12 \times 1 = 12$. Zero is therefore the **additive identity**, and one is the **multiplicative identity**: $a + 0 = a$ and $a \times 1 = a$. An opposite reverses addition: $12 + (-12) = 0$, or generally $a + (-a) = 0$. A **reciprocal** is a number that multiplies a nonzero number back to one. For example, $4 \times \frac{1}{4} = 1$ and $\frac{3}{4} \times \frac{4}{3} = 1$. In general, $a \times \frac{1}{a} = 1$ for nonzero $a$. These inverse relationships support useful checks on subtraction and division.

Subtraction and division are not commutative: $12 - 3 \neq 3 - 12$ and $12 \div 3 \neq 3 \div 12$. Rearrange an expression only when a valid property licenses the move.

#### 16.8.2.8. Estimate, calculate, inspect

A machine should enter after you know what you are asking.

Before calculating Re'u's new flock total, we can estimate:

- the starting count is 360;
- births add 150;
- all removals together are a little more than 60;
- the result should therefore be a little less than 450 and greater than 360.

A calculator returns

$$360 - 24 - 12 + 150 - 30 = 444.$$

That result lies in the expected range. We then inspect the units and the groups being counted. It is a count of currently present animals, not a count of original entrusted animals, adult animals, or animals owed at the next reckoning.

The calculator checked the arithmetic. It did not tell us which of those counts we needed.

#### 16.8.2.9. Practice and transfer

1. A process begins the day with 85 open cases, receives 27, and closes 34. Write an expression for the ending count and estimate whether it will be above or below 85 before calculating.
2. Explain the difference between "the backlog is 8" and "the backlog changed by −8."
3. Evaluate $6 + 4 \times 5$ and $(6 + 4) \times 5$.
4. Use the distributive property to compute $17 \times 12$ as $17(10 + 2)$.
5. Classify each as an expression, a true statement, a false statement, malformed notation, or an undefined expression: $12 + 7$; $12 + 7 = 19$; $12 + 7 = 20$; $12 + \div 7$; $12 \div 0$.
6. A machine reports that $360 - 24 - 12 + 150 - 30 = 4{,}440$. Give two independent reasons to reject the result before recomputing it.
7. Explain why $3 - 5 = -2$ and $1 \div 2 = \frac{1}{2}$ are defined even though their results are not, respectively, a nonnegative whole number and an integer.

Check Your Reasoning

1. $85 + 27 - 34 = 78$. Because 34 departures exceed 27 arrivals by 7, the ending count should be 7 below 85.
2. The first gives a level: eight cases are currently in the backlog. The second gives a change: the backlog decreased by eight from some prior level.
3. $6 + 4 \times 5 = 6 + 20 = 26$; $(6 + 4) \times 5 = 10 \times 5 = 50$.
4. $17(10 + 2) = 170 + 34 = 204$.
5. Expression; true statement; false statement; malformed notation; undefined expression.

6. The reported value is an order of magnitude too large, and the net change should be only +84, so the result should be near 444 rather than 4,440.
7. Each operation has a value when the available numbers expand: the first result is admitted when negative numbers are available, and the second when fractions are available. Leaving the selected number set is not the same as dividing by zero or otherwise failing to define a value.

### 16.8.3. Third Reckoning: Fractions, Rates, and Honest Approximation

#### 16.8.3.1. Try before reading

The story offers three possible wholes: the 360 animals originally entrusted, the 444 animals presently counted, and a weighted total based on differing feed needs. What question can each whole answer? Why can no one of them replace the other two?

Before calculating, decide whether $9{,}000/(444 \times 30)$ should be greater than or less than 1. What does your answer mean in the story's units?

The reserve problem forces several familiar forms of arithmetic to work together. They are related, but they are not interchangeable.

#### 16.8.3.2. Fractions name numbers

A fraction such as

$$\frac{3}{4}$$

is a number. It may describe three of four equal parts, the result of $3 \div 4$, a ratio of three to four, or a scaling factor that makes a quantity three-fourths as large.

The denominator, which cannot be zero, names how many equal parts make the selected whole. The numerator names how many of those parts are under consideration.

The phrase "the selected whole" matters. Three-fourths of a *sila*, three-fourths of the reserve, and three-fourths of the animals are different quantities even though the fraction is the same.

Equivalent fractions name the same number:

$$\frac{3}{4} = \frac{6}{8} = \frac{75}{100} = 0.75.$$

Multiplying numerator and denominator by the same nonzero number changes the representation without changing the value.

#### 16.8.3.3. Arithmetic with fractions

To add fractions with the same denominator, add their numerators:

$$\frac{1}{4} + \frac{2}{4} = \frac{3}{4}.$$

For different denominators, first express the fractions in compatible parts. Half a measure contains two quarter-measures. Those two quarters and three more quarters make five quarters, or one whole measure and one quarter:

$$\frac{1}{2} + \frac{3}{4} = \frac{2}{4} + \frac{3}{4} = \frac{5}{4} = 1\frac{1}{4}.$$

To find three-fourths of two-thirds of a measure, divide the whole measure into three equal parts and keep two. Now divide each third into four equal parts. Each small part is one-twelfth of the whole, and the two thirds contain eight such parts. Take three of the four small parts in each of those two thirds: six twelfths remain, which is half the original measure.

The denominator counts $3 \times 4 = 12$ equal parts in the whole. The numerator counts $2 \times 3 = 6$ selected parts. That is why multiplying fractions multiplies both their numerators and their denominators:

$$\frac{3}{4} \times \frac{2}{3} = \frac{3 \times 2}{4 \times 3} = \frac{6}{12} = \frac{1}{2}.$$

A common error is to multiply the numerators while leaving the first denominator unchanged, producing $\frac{6}{4}$. The neatness of the written line does not protect it from that mistake. Inspect the operation, not the typography.

How many half-measures fit into three-fourths of a measure? One half-measure uses two quarters and leaves one quarter. That remaining quarter is half of another half-measure. The answer is one and one-half half-measures.

We can express the same comparison as a fraction divided by a fraction. Multiplying both quantities by two turns the half-measure into one and the three-fourths measure into three-halves. Their ratio stays the same:

$$\frac{3/4}{1/2} = \frac{(3/4)\times 2}{(1/2)\times 2} = \frac{3/2}{1} = \frac{3}{2}.$$

Two is the reciprocal of one-half because their product is one. The same method works for any nonzero fraction: multiply both quantities by the divisor's reciprocal so the divisor becomes one. Division therefore multiplies by the reciprocal:

$$\frac{3}{4} \div \frac{1}{2} = \frac{3}{4} \times \frac{2}{1} = \frac{3}{2}.$$

It does not license division by zero, because zero has no reciprocal.

#### 16.8.3.4. Decimals may terminate or repeat

Decimal places extend the grouping rule to the right of the ones place. The first place after the decimal point counts tenths, the next counts hundredths, and the next counts thousandths. Each is one-tenth the size of the place to its left. Thus 0.75 means seven tenths and five hundredths, or seventy-five hundredths:

$$0.75 = \frac{7}{10} + \frac{5}{100} = \frac{75}{100} = \frac{3}{4}.$$

Some fractions have terminating decimal representations:

$$\frac{3}{4} = 0.75, \qquad \frac{1}{2} = 0.5.$$

Others repeat:

$$\frac{5}{6} = 0.8333\ldots,$$

$$\frac{25}{37} = 0.675675675\ldots.$$

A repeating decimal is not inherently approximate. The notation $0.\overline{675}$ represents the exact repeating value. Writing 0.68 instead is an approximation.

Base matters here. A fraction may terminate in one base and repeat in another. One-third can be written as exactly twenty sixtieths because $\frac{1}{3} = \frac{20}{60}$. In decimal notation the same number is 0.333 ...: no finite string of tenths, hundredths, and smaller decimal places gives it exactly. The number has not changed; the representational system has.

#### 16.8.3.5. Ratios, rates, and proportions

A **ratio** compares two quantities by division. After the removals, the ratio of lambs to all animals is

$$120 : 420$$

or

$$\frac{120}{420} = \frac{2}{7}.$$

A **rate** is a ratio that compares quantities with different units. The emergency feeding plan uses rates such as

$$1 \frac{\text{sila}}{\text{feeding unit} \cdot \text{day}}.$$

Multiplying by feeding units and days cancels those units:

$$330 \text{ feeding units} \times 30 \text{ days} \times 1 \frac{\text{sila}}{\text{feeding unit} \cdot \text{day}} = 9{,}900 \text{ sila}.$$

Unit cancellation is not merely a classroom ritual. It is a check on whether the calculation represents the proposed relationship.

A **proportion** states that two ratios are equal. If 330 feeding units require 330 *sila* for one day at the selected rate, thirty days require thirty times that quantity:

$$330 \text{ sila per day} \times 30 \text{ days} = 9{,}900 \text{ sila}.$$

We can check that the daily rate stays the same by comparing the two ratios:

$$\frac{330 \text{ sila}}{1 \text{ day}} = \frac{9{,}900 \text{ sila}}{30 \text{ days}}.$$

Do not set up a proportion until you have reason to believe the rate remains constant over the range in question. Thirty days of emergency planning are not evidence that animal need remains constant through every season.

#### 16.8.3.6. Percentages and percentage change

A percentage expresses a ratio per hundred. To write a ratio as a percentage, express the same share with a denominator of one hundred.

Of 150 lambs born, 120 survived the first dangerous weeks:

$$\frac{120}{150} = \frac{4}{5} = \frac{80}{100} = 0.8 = 80\%.$$

Eighty percent means eighty out of every hundred in the same proportion, not that only eighty lambs survived.

The corresponding early loss percentage was

$$\frac{30}{150} = 0.2 = 20\%.$$

The percentages add to 100% because every lamb in this particular group is classified as surviving or not surviving through the defined interval.

Percentage change compares a change with a nonzero starting value:

$$\text{percentage change} = \frac{\text{new} - \text{old}}{\text{old}} \times 100\%.$$

If the old value is zero, ordinary percentage change is undefined because the formula would divide by zero. Report the numerical change and its context instead of calling it an infinite percentage increase.

The flock changed from 360 at entrustment to 444 present animals before the allocation decision:

$$\frac{444 - 360}{360} \times 100\% = \frac{84}{360} \times 100\% \approx 23.3\%.$$

That does not mean the entrusted flock reproduced by 23.3 percent. The change also contains transfers, adult deaths, births, and lamb deaths. The arithmetic is correct; the causal description would be wrong.

Always name the denominator. A claim that defects fell “by 20 percent” is incomplete until we know the starting defect count, the time interval, and whether the percentage describes items, opportunities, orders, or something else.

#### 16.8.3.7. Exactness, approximation, and rounding

An exact value preserves the represented relationship without rounding:

$$\frac{25}{37} \text{ sila per animal per day.}$$

An approximation makes the value easier to communicate or use at an available level of precision:

$$\frac{25}{37} \approx 0.68.$$

The approximation is not a defective version of the truth. It is a different representation with an error small enough for some purpose. The phrase "for some purpose" carries the burden.

Rounding an intermediate rate and then multiplying it across many animals and days can accumulate error. Whenever possible, preserve the exact fraction or additional digits during the calculation and round the final result to a precision justified by the measurement and decision.

Estimation provides an independent check. The weighted plan after removing twenty-four adults uses roughly 400 animals, roughly one feeding unit or less per animal, for 30 days. A result near 10,000 *sila* is plausible. A result of 990 or 99,000 would demand investigation before anyone opened a jar.

#### 16.8.3.8. Use a machine without surrendering the question

Enter the plan into a calculator or spreadsheet as separate rows:

| Class | Count | Feeding units per animal | Days | Required *sila* |
|---|---|---|---|---|
| Breeding or nursing ewes | 180 | 1 | 30 | 5,400 |
| Other adults | 120 | 0.75 | 30 | 2,700 |
| Lambs | 120 | 0.50 | 30 | 1,800 |
| **Total** | **420** | | | **9,900** |

The machine should calculate the final column as

$$\text{animals} \times \frac{\text{feeding units}}{\text{animal}} \times \text{days} \times 1 \, \frac{\text{sila}}{\text{feeding unit} \cdot \text{day}}.$$

Then inspect:

- Do the animal counts sum to 420?
- Do the units cancel to *sila*?
- Is 9,900 close to the estimate?
- Does the table show the groups and the choices behind their weights?
- Would changing a weight alter a biological fact, a policy choice, or both?

The final question cannot be delegated to the spreadsheet.

#### 16.8.3.9. Practice and transfer

1. Convert 7 *gur*, 2 *barig*, and 3 *ban* to *sila* using 1 gur = 300 sila, 1 barig = 60 sila, and 1 ban = 10 sila.
2. A reserve of 9,000 *sila* is divided equally over 444 animals and 30 days. Without calculating the exact rate, explain why the result must be less than 1 *sila* per animal per day.
3. Compute the exact rate and identify its repeating decimal pattern.
4. After the removals, what percentage of the 420 animals are lambs?
5. Of 120 completed cases in one cohort, 84 were completed within the promised time and the rest were late. Find the on-time percentage and the late percentage.
6. A defect count falls from 50 to 35. Find the change and the percentage change.
7. A spreadsheet multiplies 120 lambs by 0.5 feeding unit per lamb but omits the 30-day factor. What unit will its result have, and why is that not the required total?
8. Explain why releasing barley according to the 360-animal tablet and releasing it according to the 444 present animals are not merely two methods for answering the same question.
9. Compute $\frac{1}{2}+\frac{3}{4}$ and explain why the denominators must name compatible parts before the numerators are added.
10. Compute $\frac{3}{4}\times\frac{2}{3}$ and simplify the result.
11. Compute $\frac{3}{4}\div\frac{1}{2}$ and explain what the quotient means as a comparison of the two quantities.
12. Keep the fourth plan's counts and relative weights, but propose 0.9 *sila* per feeding unit per day instead of 1. Does the feeding-unit total change? How much barley does the revised thirty-day plan allocate, and does that calculation establish that the animals will receive enough?

> Check Your Reasoning
>
> 1. $7(300)+2(60)+3(10)=2{,}100+120+30=2{,}250$ *sila*.

2. One *sila* per animal per day would require $444 \times 30 = 13{,}320$ *sila*, which exceeds the 9,000-*sila* reserve.
3. $9{,}000/(444 \times 30) = 25/37 = 0.675675\ldots$, with 675 repeating.
4. $120/420 = 2/7 \approx 0.2857$, so about 28.6% are lambs.
5. $84/120 = 70\%$ within the promise; $(120 - 84)/120 = 36/120 = 30\%$ late.
6. The change is $35 - 50 = -15$ defects, a decrease of 15. The percentage change is $-15/50 \times 100\% = -30\%$.
7. The result is 60 feeding units. It lacks both the 30-day duration and the rate in *sila* per feeding unit per day, so it is not the required quantity of barley.
8. The first distributes against a historical obligation recorded at entrustment. The second distributes among currently present mouths. The unit boundary and purpose differ before the division begins.
9. $\frac{1}{2} + \frac{3}{4} = \frac{2}{4} + \frac{3}{4} = \frac{5}{4} = 1\frac{1}{4}$. Halves and quarters are not the same-sized parts, so the halves must first be expressed as quarters.
10. $\frac{3}{4} \times \frac{2}{3} = \frac{6}{12} = \frac{1}{2}$.
11. $\frac{3}{4} \div \frac{1}{2} = \frac{3}{4} \times \frac{2}{1} = \frac{3}{2} = 1\frac{1}{2}$. Three-fourths is one and one-half times as large as one-half.
12. The daily feeding-unit total remains 330 because the counts and relative weights have not changed. The revised plan allocates $330 \times 30 \times 0.9 = 8{,}910$ *sila*. That fits within the assigned 9,000 *sila*, but it does not establish that the smaller ration is adequate. Adequacy depends on the animals, grazing, weather, and the evidence behind the chosen ration.

### 16.8.4. Fourth Reckoning: Exponents, Factors, and Prime Structure

#### 16.8.4.1. Try before reading

Break 360 apart by starting with $6 \times 60$. Then start again with $18 \times 20$. If you continue until every factor is prime, what do you expect to remain unchanged?

Would writing the number as $(6, 0)_{60}$ instead of $360_{10}$ change those factors? Make a prediction before reading the theorem that answers both questions.

#### 16.8.4.2. Repeated multiplication

Some factors repeated. Repeated multiplication can be written compactly with an **exponent**:

$$2 \times 2 \times 2 = 2^3,$$

$$3 \times 3 = 3^2.$$

In $2^3$, 2 is the **base** and 3 is the **exponent**. The expression means three factors of 2:

$$2^3 = 8.$$

This use of the word *base* differs from the base of a numeral system. In $2^3$, the base is the repeated factor. In base-ten notation, the base determines place values. Context keeps the two uses apart.

Follow the powers of two downward. Removing one factor of two divides the value by two:

$$2^3 = 8, \qquad 2^2 = 4, \qquad 2^1 = 2.$$

If we continue the same pattern, the next value is $2 \div 2 = 1$. We therefore define $2^0 = 1$ so that decreasing the exponent by one continues to divide the value by two. It does not mean $2 \times 0$.

The same reasoning works for any nonzero base. For example, $3^2 = 9$, $3^1 = 3$, and $3^0 = 1$: divide by three at each step. Writing $a$ for the chosen base, $a^1 = a$ and $a^0 = 1$ for nonzero $a$. The pattern cannot make that last step for a zero base because it would require division by zero.

Division of powers now gives another way to check the rule. In $2^3/2^2$, two factors of two in the numerator cancel the two in the denominator, leaving one:

$$\frac{2 \times 2 \times 2}{2 \times 2} = 2 = 2^{3-2}.$$

Dividing $2^3$ by itself cancels all three factors and gives 1. Subtracting the exponents gives zero, so $2^0 = 1$ preserves both relationships. More generally, for nonzero $a$:

$$\frac{a^3}{a^3} = 1 = a^{3-3} = a^0.$$

#### 16.8.4.3. Factors and primes

A positive integer $a$ is a **factor** of a positive integer $n$ if some positive integer $b$ satisfies

$$a \times b = n.$$

Because

$$12 \times 30 = 360,$$

both 12 and 30 are factors of 360.

A positive integer greater than 1 is **prime** if its only positive factors are 1 and itself. The first primes are

$$2, 3, 5, 7, 11, 13, 17, 19, \ldots$$

A positive integer greater than 1 that is not prime is **composite**. It can be expressed as a product of smaller positive integers.

The number 1 is neither prime nor composite. If 1 were prime, we could insert as many factors of 1 as we liked into every prime factorization and destroy the useful meaning of uniqueness.

Zero is not prime either. It is divisible by every nonzero integer and does not have the factor structure required by the definition.

#### 16.8.4.4. Factor 360

Begin with any nontrivial factor pair:

$$360 = 6 \times 60.$$

Then factor each composite part:

$$6 = 2 \times 3,$$

$$60 = 6 \times 10 = (2 \times 3)(2 \times 5).$$

Combining all prime factors gives

$$360 = 2 \times 3 \times 2 \times 3 \times 2 \times 5.$$

Reorder the factors and use exponents:

$$\boxed{360 = 2^3 \times 3^2 \times 5}.$$

Check:

$$2^3 \times 3^2 \times 5 = 8 \times 9 \times 5 = 72 \times 5 = 360.$$

Starting with another factor pair leads to the same prime factors. For example:

$$360 = 18 \times 20 = (2 \times 3^2)(2^2 \times 5) = 2^3 \times 3^2 \times 5.$$

Both routes produce the same prime factors.

#### 16.8.4.5. The Fundamental Theorem of Arithmetic

The **Fundamental Theorem of Arithmetic** states that every integer greater than 1 can be written as a product of primes, and that this prime factorization is unique apart from the order of the factors. You do not need to prove the theorem here. You need to recognize what it guarantees and be able to check a modest factorization.

For 360, the order can change:

$$2^3 \times 3^2 \times 5 = 5 \times 3 \times 2 \times 3 \times 2 \times 2.$$

Those are not competing factorizations. They contain the same prime factors with the same multiplicities.

The theorem is a claim about positive integers, not about the numeral marks used to represent them. The number 360 may appear as

$$360_{10}$$

or

$$(6, 0)_{60}.$$

Its representation changes with the numeral base. Its prime factorization remains

$$2^3 \times 3^2 \times 5.$$

Numerals depend on a representational system; prime factorization belongs to the integer represented.

It also explains why sixty is convenient for division.

$$60 = 2^2 \times 3 \times 5.$$

Because its prime factors include 2, 3, and 5, sixty has many divisors:

$$1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60.$$

That makes halves, thirds, quarters, fifths, sixths, tenths, twelfths, and other common shares fit neatly into sexagesimal work. It is one practical virtue of the system, not a complete history of why people developed and retained it.

#### 16.8.4.6. A practical factoring method

For a modest positive integer:

1. divide by the smallest prime that works;
2. divide the quotient again by the smallest prime that works;
3. continue until the quotient is prime; and
4. collect repeated factors with exponents.

Factor 444:

$$444 = 2 \times 222 = 2 \times 2 \times 111 = 2^2 \times 3 \times 37.$$

The number 37 is prime, so the process stops.

Check with multiplication:

$$2^2 \times 3 \times 37 = 4 \times 3 \times 37 = 12 \times 37 = 444.$$

A machine can check this quickly. It can also generate a prime factorization. You still need to know what the output claims and whether the input was the intended positive integer.

Factoring 444 when the question concerned 420 is flawless work on the wrong number.

#### 16.8.5. Responsible calculation

The arithmetic chapter can now be reduced to a short procedure, although living by it is not always short.

Before computation:

1. **Name the thing or event.** What is being observed?
2. **Define one.** What makes one member, occurrence, interval, or opportunity?
3. **Set the boundary.** Which place, population, process step, and time window belong?
4. **Preserve the unit.** What does each value measure or count?
5. **Choose the relationship.** Does the situation call for a sum, difference, product, quotient, rate, proportion, or percentage?
6. **Estimate.** What sign, order of magnitude, and rough range should a plausible answer have?

After computation:

7. **Check the arithmetic.** Recompute, reverse the operation, or use another tool.
8. **Check the representation.** Did rounding, grouping, missing data, or an incorrect denominator change the claim?
9. **Interpret narrowly.** State what the result actually establishes.
10. **Disclose what it omits.** Name material assumptions and unrepresented costs.
11. **Decide.** Arithmetic can clarify alternatives and consequences; it cannot accept responsibility for the action.

This is not a ceremonial checklist to paste after every spreadsheet. It is a compact description of habits that become faster with use.

#### 16.8.6. A machine-assisted reconciliation

Consider a four-week intake process. The working table contains these counts:

| Week | Applications received | Applications completed | Rework events | Applications first counted with rework |
|---|---|---|---|---|
| 1 | 48 | 42 | 6 | 5 |
| 2 | 52 | 50 | 4 | 3 |
| 3 | 45 | 44 | 5 | 4 |
| 4 | 55 | 49 | 6 | 4 |
| **Total** | **200** | **185** | **21** | **16** |

The totals are easy for a spreadsheet. Before entering a formula, define the columns.

- One application received means one application assigned a unique identifier and accepted into this process during the week.
- One application completed means one application that met the selected completion rule during the week, whether it arrived that week or earlier.
- One rework event means one recorded return from a later step to an earlier step.
- One application first counted with rework means one distinct application whose first recorded return **within this four-week window** occurred during that week. Count it in that week's final column only, even if it has further rework in later weeks.

An application returning in both Week 1 and Week 3 contributes rework events in both weeks, but appears in the final column only in Week 1. That rule makes the weekly application counts nonoverlapping, so their sum is 16 distinct applications across the four weeks. If each week instead counted every application with rework that week, adding the weekly counts would not establish a distinct count for the whole period. We would have to count unique application identifiers across the underlying records.

Those definitions reveal why received applications should not be subtracted from completed applications week by week and called defects. The columns describe flow across a boundary, not two classifications of the same weekly group. A completion may come from an earlier week's intake.

Start with a rough prediction. Intake is near 50 per week, so the four-week total should be near 200. Completion is a little lower, so its total should be somewhat below 200. If the spreadsheet reports 2,000 received or 18.5 completed, inspect the entries and formulas before interpreting the result.

In a spreadsheet, the total received might be calculated with a formula such as:

```
=SUM(B2:B5)
```

The leading = tells the spreadsheet to evaluate an expression. `SUM` applies addition to a selected range. The result is 200 only if the intended data occupy those cells and every row uses the same unit. A correct formula over the wrong range is the spreadsheet equivalent of a flawless count at the wrong gate.

Now reconcile the process balance. Suppose 18 applications were already open at the beginning of Week 1, and a direct count finds 31 open after Week 4. If receipts were the only additions and completions the only removals, the expected closing balance would be

$$18 + 200 - 185 = 33.$$

The direct observation says 31. The arithmetic has not failed; the account is incomplete. Investigation finds two applications withdrawn by their submitters and omitted from the first table. The reconciled balance is

$$18 + 200 - 185 - 2 = 31.$$

Write this as a general balance relationship:

$$\text{closing work} = \text{opening work} + \text{entries} - \text{exits}.$$

The words *entries* and *exits* should be replaced by the actual kinds of entry and exit in the process. A balance equation is useful because an unexplained difference becomes visible. It does not tell you whether the difference came from an omitted kind of work, a duplicate record, a changed definition, or a bad observation. That requires investigation.

The same table supports several different ratios. Over four weeks, the process recorded 21 rework events across 200 received applications:

$$\frac{21 \text{ rework events}}{200 \text{ received applications}} = 0.105 \frac{\text{events}}{\text{application}}.$$

That is 0.105 recorded rework events per application received during this window. It is a volume-normalized event rate, not necessarily the average number of rework events experienced by those 200 applications. That interpretation would require the events in the numerator to come from the same application cohort as the denominator. It is also not the percentage of

applications with rework because one application can generate more than one event.

The distinct-application count supports a different calculation:

$$\frac{16 \text{ applications with rework}}{200 \text{ received applications}} = 0.08 = 8\%.$$

This percentage also needs a cohort rule. If some of the 16 applications arrived before Week 1, while some of the 200 new applications will experience rework after Week 4, the numerator and denominator do not describe the same population. The calculation is numerically correct but conceptually misaligned. A defensible cohort measure might follow the 200 applications received in the four-week window until each completes, then count how many ever experienced rework.

The completed total gives an observed average:

$$\frac{185 \text{ completions}}{4 \text{ weeks}} = 46.25 \frac{\text{completions}}{\text{week}}.$$

That rate summarizes the selected four weeks. It does not mean the process completed one quarter of an application, that every week produced 46.25 completions, or that future capacity is exactly 46.25 per week. The decimal belongs to the average, not to a physical fragment of an application.

A reliable workbook should preserve at least four layers:

1. the source observations;
2. the definitions and units attached to them;
3. the formulas that transform them; and
4. the reported values, including any rounding or formatting.

Do not type a displayed total back over its formula. Do not convert a fraction to a rounded percentage and later treat the displayed percentage as the original data. Where the decision matters, keep the count behind the rate so another person can reconstruct it.

Finally, perform a deliberate independent check. Add the four weekly receipt counts in a calculator or reverse the balance equation to solve for openings. If you have the underlying rework records with application identifiers, count the unique identifiers across all four weeks to check the distinct-application total. The aggregate table alone cannot show whether an identifier was counted twice. The purpose is not distrust of machines. It is to avoid asking one unnoticed error to certify itself.

### 16.8.7. Final practice

1. Convert $(8, 6)_{60}$ to decimal notation.
2. Convert $444_{10}$ to modern explicit base-sixty notation.
3. Factor 420 into primes.
4. Factor 384 into primes and verify by multiplication.
5. Evaluate $2^5$ and explain what the exponent means.
6. Continue the pattern $3^3 = 27$, $3^2 = 9$, $3^1 = 3$ by dividing by three at each step. What value should $3^0$ have, and why is that different from $3 \times 0$?
7. Explain why the prime factorization of 360 does not change when the number is written in base sixty.
8. A process records 72 defects across 600 inspected units. Find the defects-per-unit rate. Explain why this is not necessarily the percentage of defective units.
9. A weekly count rises from 80 cases to 100 cases. Find the numerical and percentage changes.
10. A calculator displays 2.666666667 for 8/3. Distinguish the exact value from the displayed approximation.
11. A team reports average handling time in minutes but enters one day's data in seconds. Why can the arithmetic be internally correct while the result is unusable?
12. Write a short answer to the question "What did you make one?" for each of these measures: completed orders, defect percentage, waiting time, and customer complaints.

Check Your Reasoning

1. $(8, 6)_{60} = 8 \times 60 + 6 = 486_{10}$.
2. $444 = 7 \times 60 + 24$, so $444_{10} = (7, 24)_{60}$.
3. $420 = 42 \times 10 = (2 \times 3 \times 7)(2 \times 5) = 2^2 \times 3 \times 5 \times 7$.
4. $384 = 128 \times 3 = 2^7 \times 3$. Check: $128 \times 3 = 384$.
5. $2^5 = 2 \times 2 \times 2 \times 2 \times 2 = 32$; the exponent 5 says that five factors of 2 are multiplied.
6. The next value is $3 \div 3 = 1$, so $3^0 = 1$ continues the pattern. The zero is an exponent; it does not instruct us to multiply by zero. The product $3 \times 0$ is zero, a different calculation.
7. Numeral base changes the marks and place values used to represent the integer, not the integer's factors.
8. $72/600 = 0.12$ defects per inspected unit. A single unit may contain more than one defect, so the count of defective units may be smaller than the count of defects.
9. The numerical change is $100 - 80 = 20$ cases, an increase of 20. The percentage change is $20/80 \times 100\% = 25\%$.

10. The exact value is $8/3 = 2.\overline{6}$. The display rounds that repeating decimal to a finite number of digits.
11. The machine may correctly combine the entered numbers while combining incompatible units. Convert all values to one unit before aggregation.
12. Answers will vary, but each must define the member or event, the boundary, and the time window where relevant. For complaints, for example: one complaint might mean one submitted complaint record associated with one order during the selected month, rather than every issue mentioned inside the record.

### 16.9. Chapter Summary

- A thing, quantity, number, numeral, and unit are related but distinct.
- Counting requires a criterion for treating something as one member of a collection.
- Zero may describe an empty quantity or preserve a place in positional notation.
- A number can have different numeral representations in different bases.
- Arithmetic operations transform represented quantities; their meaning depends on context and compatible units.
- Signed numbers can describe direction and change without turning physical things into "negative objects."
- Fractions are numbers and can also express division, ratios, rates, and scaling.
- Percentages are ratios per hundred, and percentage change requires an explicit, nonzero starting value.
- Exact values and approximations serve different purposes; rounding should follow the needed use and precision.
- A machine can perform and check computation but cannot choose the unit, boundary, denominator, or decision.
- Every integer greater than 1 has a prime factorization unique apart from factor order.
- A correct calculation can expose the consequences of a frame without selecting the frame or accepting responsibility for what follows.

### 16.10. Historical Note and Further Reading

This chapter uses Reed Gate, a fictional settlement in the rural orbit of Nippur around 1800 BCE. Its people and conflict are invented. The following sources support the material and institutional frame and also mark its limits:

- The University of Chicago's *Nippur Neighborhoods* treats the domestic, social, and economic world of Old Babylonian Nippur.
- ORACC's Old Babylonian Model Contracts provides editions and translations of scribal model contracts involving barley, property, loans, witnesses, and related transactions.
- ORACC's lexical entry for *nakamtum*, storehouse attests Old Babylonian Nippur terms for a storehouse, its doorkeeper, an overseer of the storehouse, and a storehouse official. "Storehouse overseer" therefore has a historical basis; Abum-ilum's precise duties, Ili-iddinam's assignment, and their intersecting authority are narrative constructions.
- ORACC's lexical entries for shepherd, sheepfold, and lamb document relevant Old Babylonian vocabulary and classifications.
- The University of Chicago's *Assyrian Dictionary*, Š, Part III, entry *šūši*, documents the word for sixty, including Old Babylonian examples. ORACC's Old Babylonian mathematical glossary records *meʾatu*, hundred, and its *limum* entry records thousand, Akkadian *līmu*, with Old Babylonian attestations. The characters' familiarity with hundreds and thousands is distinct from Nisaba's introduction of modern decimal notation.
- ORACC's account of Nisaba/Nidaba describes the goddess's associations with grain, writing, and scribal culture and her cultic connection to Nippur.
- ORACC's account of Mesopotamian metrology and the Cuneiform Digital Library Initiative's survey of Babylonian mathematics describe the sexagesimal place-value tradition behind Ili-iddinam's work. Nisaba's modern decimal digits, her explicit zero placeholder, and the notation $(6, 0)_{60}$ are deliberate anachronisms, not reproductions of an Old Babylonian tablet.
- The University of Hamburg's overview of Mesopotamian calendars and the Cuneiform Digital Library Initiative's study of the Nippur calendar describe locally variable lunisolar calendars tied to seasons and festivals. The unnamed summer feast and the children's wreath are narrative constructions, not a reconstruction of an attested Nippur solstice festival; an Old Babylonian term for a garland or tiara is attested.
- The Metropolitan Museum's Old Babylonian bronze figure provides a material comparison for small period bronze work. Nisaba's scale, seamless construction, movement, speech, and changes of apparent weight are intentionally impossible.
- Stephanie Dalley's study of Old Babylonian dowries and the University of Tübingen's inheritance research illustrate why a widow's claims must be specified rather than inferred from a generic statement about women's status.

Reed Gate's name, the storehouse's exact relationship to an unnamed Nippur temple household, the feeding-unit weights, division of the reserve,

household arrangement, temple decree, pledge, and final outcome are narrative constructions. They should not be treated as a reconstruction of one surviving Old Babylonian practice.

Part V.

# Exercise Answers and Guidance

# 17.

# Solutions to White Belt Exercises

This appendix collects answers and acceptance guidance for the public White Belt chapter and capstone exercises. Open-ended responses are judged by the stated criteria rather than one prescribed answer. The operational chapter reviews and cumulative White Belt assessment are delivered separately through the LMS and are not included here.

## 17.1. Lean Foundations

### 17.1.1. Exercise 1

**1.** Understand In your own words, explain how the four foundational concepts — firm, customer, value, and process — are related. Your answer should be a short paragraph that could serve as an explanation to a colleague who has never studied process improvement.

**Worked solution**

A firm is an organized entity whose reason for existing is to provide value to its customers. The customer is the person or organization that receives the benefit of the firm's services or the use of its goods, whether that customer or another party pays. Value is the worth those goods or services have from the customer's point of view. The mechanism by which the firm actually delivers that value is a process: a repeatable set of steps that transforms inputs into a product or service. Improve the process, and you improve the firm's ability to create value for the customer.

### 17.1.2. Exercise 2

**2.** Understand Using the Wash n' Fold case, identify the firm, the customer, the core value the customer receives, and the process being improved. Write your answer in 3–4 sentences.

**Worked solution**

The firm is the Wash n' Fold business. The customer is the person who receives the clean, folded laundry and the benefit of reliable return, regardless of who pays. The core value is clean, correctly handled laundry returned when promised. The process being improved begins when the customer drops off an order and ends when the completed order is staged and the customer is notified that it is ready.

### 17.1.3. Exercise 3

**3.** Understand In your own words, define a **process run** and explain why observing multiple runs is better than observing only one.

**Worked solution**

A process run is one complete instance of the process moving from its start event to its stop event for one thing being followed. Observing multiple runs is better because one run may be unusual, while repeated runs let you see typical timing, variation, and recurring trouble spots.

### 17.1.4. Exercise 4

**4.** Understand Think about a process you participate in regularly — at work, at school, or in daily life. In one or two sentences, describe the process and identify who the customer is. Then: does the firm or organization running that process seem to understand what you, as the customer, actually value? Give one piece of evidence for your view.

**Worked solution**

Answers will vary. Look for (1) a clearly named process with a plausible customer identified, (2) a specific value claim—not just "good service," but something concrete the customer cares about—and (3) observable evidence rather than opinion alone. A strong answer may find alignment or tension between the firm's priorities and customer value, but its conclusion should follow from the evidence given.

### 17.1.5. Exercise 5

**5.** Understand In plain language, explain what process variants are and why we care about them for process improvement work.

**Worked solution**

Process variants are recurring alternative paths through the same process boundary. We document them so that we can distinguish necessary paths from inconsistent work, exceptions, and rework instead of treating every difference as either harmless or defective.

### 17.1.6. Exercise 6

**6.** Understand Why can a process look busy internally while still underperforming for customers? Answer using value and process concepts from this chapter.

**Worked solution**

Visible activity does not guarantee that the process is producing the outcome the customer values. People may be working hard on queues, rework, handoffs, or internal requirements while the customer still waits for the promised product or service.

### 17.1.7. Exercise 7

**7.** Understand Write a two-sentence explanation of why process thinking is a prerequisite for meaningful improvement work.

**Worked solution**

Process thinking connects an outcome to the sequence of work that produces it. Without that connection, a team cannot locate delays, compare variants, or test whether a change improved the mechanism responsible for the result.

#### 17.1.8. Exercise 8

**8.** Understand Explain why continuous improvement and respect for people must operate together. Give one example of an improvement practice that supports both foundations.

**Worked solution**

Continuous improvement without respect can turn efficiency into coercion or exploitation; respect without improvement can leave preventable burdens and failures in place. For example, asking the people who perform the work to identify recurring obstacles, testing one of their proposed changes, and reviewing the result with them uses their knowledge while improving the process.

### 17.2. The Five Principles of Lean

#### 17.2.1. Exercise 1

**1.** Understand Explain why the five Lean principles are presented in this order: define value, map the process, create flow, establish pull, and pursue perfection. For each transition, state why the next principle is needed.

**Worked solution**

A strong explanation follows the logic of the sequence. We define value first so that improvement has a customer-centered purpose. We then map the process because we need to see how the work currently creates—or fails to create—that value. Once the work is visible, we can improve flow by reducing waiting, interruption, rework, and unnecessary batching. After improving movement, we establish pull so that demand and downstream capacity govern when work enters the process. Finally, we pursue perfection because every change creates a new current state and customer needs and operating conditions continue to change.

### 17.2.2. Exercise 2

**2.** (Understand) Classify the flow and release logic in each process.

- A bakery decides each morning to produce 200 loaves from its sales forecast. Once production begins, each loaf moves from shaping to proofing to baking with little waiting.
- A custom-print shop begins a job only after receiving a customer order, but then holds that job until 50 similar orders accumulate for printing.

For each process, state whether it demonstrates relatively good or poor flow and whether its release is push or pull.

**Worked solution**

The bakery has relatively good flow after production begins because units move with little waiting, but its release is **push** because the forecast authorizes production. The print shop uses **pull** at the initial release because a customer order authorizes the job, but it has poor flow afterward because the job waits for a batch to accumulate. The examples show why flow and pull must be diagnosed separately.

### 17.2.3. Exercise 3

**3.** (Analyze) A hospital receives referrals throughout the day. Intake staff enter each referral immediately, but at 4 p.m. the system releases every accumulated referral to the scheduling queue, whether or not the scheduling team has capacity. Schedulers begin contacting patients the following morning.

Identify one flow problem and one pull-control problem. State what evidence you would collect before recommending a change.

**Worked solution**

The referrals wait after intake and again overnight, so the batch release interrupts flow and lengthens response time. At the intake boundary, an actual referral creates demand; at the scheduling boundary, however, the clock pushes the entire accumulated batch forward regardless of downstream capacity. Useful evidence would include referral arrival times, time spent waiting at each stage, scheduler capacity by time of day, queue size, time to first patient contact, and the frequency of incomplete or reworked referrals. That evidence would show whether smaller and more frequent releases could improve flow without overloading the schedulers.

### 17.2.4. Exercise 4

**4.** Analyze Apply all five Lean principles to the Wash n' Fold case. For each principle, write one diagnostic question and identify one piece of evidence you would seek before proposing a countermeasure.

**Worked solution**

Answers will vary, but a strong response connects each principle to observable evidence:

1. **Define value:** What turnaround time, accuracy, and garment care do customers expect? Examine customer interviews, complaints, repeat business, and promised service levels.
2. **Map the process:** What path does an order follow from drop-off until it is staged and the customer is notified that it is ready? Observe the work and record actual steps, handoffs, queues, decisions, and rework.
3. **Create flow:** Where does an order stop after processing begins? Measure queue lengths and waiting time without assuming which step is constrained.
4. **Establish pull:** What authorizes new work to enter each step, and can downstream capacity absorb it? Compare release decisions with work-in-process levels and available capacity.
5. **Pursue perfection:** After a change, what problem or constraint remains? Compare the new results with the baseline, then select the next bounded experiment.

#### 17.2.5. Exercise 5

**5.** Apply A team has reduced the average approval time in a functioning service process from five days to two. The service now meets most customer needs, but 10 percent of requests still take more than a week. One manager declares the process fixed; another proposes replacing the entire service immediately.

Using **Pursue Perfection**, recommend a more disciplined next step.

**Worked solution**

The team should not treat the current gain as final, but the remaining delay does not by itself justify immediate wholesale replacement. It should study the delayed 10 percent, determine whether those requests share a cause, and test a bounded countermeasure. The team should consider fundamental redesign if the evidence shows that the process's basic design cannot reliably deliver what customers value. Pursuing perfection means continuing to learn, not changing everything at once.

### 17.3. A Structured Approach to Process Improvement

#### 17.3.1. Exercise 1

**1.** Apply A small dental practice reviewed 40 scheduled appointments from the previous month. In 18 of them, the patient waited more than 15 minutes past the scheduled appointment time before the hygienist began the cleaning.

Write a one-sentence problem statement for this practice. Then use your imagination to write a weak version of the same statement that violates the "no causes, no blame" rule — and briefly explain what makes it weak.

**Worked solution**

**Strong:** "Last month, 18 of 40 scheduled patients waited more than 15 minutes past their appointment time before the hygienist began the cleaning."
**Weak:** "The reception staff are slow and the room-cleaning process is disorganized, causing patient delays."
The weak version jumps to blame (reception staff) and assumes causes (disorganization) before any analysis has been done. It closes off inquiry rather than opening it.

### 17.3.2. Exercise 2

**2.** (Understand) A team is running a small improvement effort on late customer orders. For each activity below, identify whether it belongs primarily to **Plan**, **Do**, **Check**, or **Act**:

- Compare on-time completion before and after the test.
- Write a neutral problem statement tied to missed pickup times.
- Update the standard work and train staff on the new routine.
- Test a new folding schedule for one week.

**Worked solution**

- **Check:** compare on-time completion before and after the test.
- **Plan:** write a neutral problem statement tied to missed pickup times.
- **Act:** update the standard work and train staff on the new routine.
- **Do:** test a new folding schedule for one week.

### 17.3.3. Exercise 3

**3.** (Apply) Consider this goal statement: "Improve customer satisfaction soon by making operations better."

Identify at least three SMART criteria this statement fails to meet and rewrite it as a one-sentence SMART goal. For the rewrite, use the Wash n' Fold case's current promise-violation rate of approximately 28%, target of 10%, and 30-day improvement period.

**Worked solution**

It fails Specific, Measurable, and Time-bound. It also gives us too little information to judge whether the goal is achievable or relevant to the business problem. A stronger version: "Reduce same-day promise violations—the percentage of orders dropped off by 9:00 a.m. that are not ready and notified by 5:00 p.m.—from approximately 28% to 10% within 30 days."

#### 17.3.4. Exercise 4

**4.** Apply A manager wants to skip the Check phase because "everyone can already tell the process feels better." Would you accept that? Why or why not?

**Worked solution**

No. Perceived improvement is not enough; the Check phase requires before-and-after comparison using consistent baseline metrics. Skipping it risks standardizing a change that did not improve the intended measure or that created harm somewhere else in the process.

#### 17.3.5. Exercise 5

**5.** Apply A university's financial aid office processes student grant applications. Students currently wait an average of 18 business days for a decision; the office's own service standard is 10 days. The director wants to run a PDCA cycle to close that gap.

Write a one-sentence problem statement and a one-sentence SMART goal for this cycle. Keep the problem statement factual and customer-focused; make the goal specific, measurable, and time-bound.

**Worked solution**

**Problem statement:** "Grant applicants are waiting an average of 18 business days for a decision, 8 days longer than the office's stated 10-day service standard."
**SMART goal:** "Reduce the average grant-application decision time from 18 to 10 business days within 60 days."
Strong responses keep the problem statement free of causes or blame, state the customer-facing delay against the service standard, and make the goal measurable (18 → 10 days) and time-bound (60 days).

### 17.3.6. Exercise 6

**6.** (Understand) What is the difference between a *problem statement* and a *SMART goal*? A teammate drafts the following and labels it as the problem statement for their PDCA cycle:

> "We need to reduce order lead time by 20% over the next month."

Explain what is wrong with calling this a problem statement, and identify the current-state facts you would need before you could write a proper problem statement.

**Worked solution**

A problem statement describes the gap between current performance and what is expected or desired, without naming causes or solutions. A SMART goal describes the intended improvement: how much, by when.
The teammate's draft is a goal, not a problem statement — it names a target (−20%) and a deadline (one month) rather than describing the current situation.
Before writing a proper problem statement, the team needs evidence about the current lead time, the relevant customer or service requirement, who is affected, and the observable consequence of the gap. The proposed reduction alone does not supply those facts, so inventing them would make the statement look precise without making it true.

### 17.3.7. Exercise 7

**7.** Apply Records show that 24% of customer issues handled last month required a callback, while the service standard allows no more than 10%. A supervisor describes the problem this way: "Our team is careless, so customer callbacks are too high." Rewrite the supervisor's statement as a neutral problem statement using the supplied evidence.

**Worked solution**

"Last month, 24% of customer issues required a callback, exceeding the service standard of no more than 10% by 14 percentage points." This version states the observed gap and removes the unsupported claim that carelessness caused it.

### 17.3.8. Exercise 8

**8.** Understand In one sentence each, define Plan, Do, Check, and Act as used in this chapter.

**Worked solution**

- **Plan:** Define the problem and goal, map the current process, establish the baseline, and choose a bounded countermeasure to test.
- **Do:** Carry out the test under stated conditions and record what happens.
- **Check:** Compare the result with the baseline and the prediction using consistent measures.
- **Act:** Standardize what the evidence supports, revise or abandon what it does not, and frame the next cycle.

### 17.3.9. Exercise 9

**9.** Apply Draft one SMART goal for a process with current on-time completion of 62%, a desired level of 85%, and a 60-day improvement period.

**Worked solution**

"Increase on-time completion from 62% to at least 85% within 60 days." The real project would also define what counts as on time and which process boundary and cohort the percentage covers.

#### 17.3.10. Exercise 10

**10.** Apply A team has one proposed countermeasure but no baseline metric. Write two sentences describing what must be added before entering Do.

**Worked solution**

The team must define a measure tied to the problem and record its current value for a stated process boundary, cohort, and period. It must also state the result it expects from the countermeasure so that Check can compare the after-state result with both the baseline and the prediction.

#### 17.3.11. Exercise 11

**11.** Apply Write a short Check-phase evaluation note template with three fields your team must fill in.

**Worked solution**

One useful template is:

- **Baseline and prediction:** What measure did we start with, and what did we expect?
- **Observed result:** What happened under the test conditions, using the same measure and boundary?
- **Interpretation and next action:** What does the comparison support, what remains uncertain, and should we adopt, revise, or abandon the countermeasure?

## 17.4. Following the Work: Building the Event Log

### 17.4.1. Exercise 1

**1.** Understand For each statement below, label it **Observed Condition** or **Assumed Cause**:

- “Orders wait 20 to 40 minutes before final packaging starts.”
- “The staff is disorganized during shift handoff.”
- “Three orders returned to the labeling step in one afternoon.”

**Worked solution**

- Observed Condition
- Assumed Cause
- Observed Condition

### 17.4.2. Exercise 2

**2.** Apply Choose a process you know and draft a mapping boundary in two sentences:

1. sentence one defines start and stop events from the customer point of view,
2. sentence two defines the unit of analysis.

**Worked solution**

Answers vary. Look for clear customer-centered boundaries and one explicit analysis unit (for example, one invoice, one service ticket, one order).

### 17.4.3. Exercise 3

**3.** Analyze A team produced this observation snippet:

- Intake: 4 min work, 0 min wait
- Verification: 6 min work, 18 min wait
- Approval: 3 min work, 22 min wait

Which two delays should be prioritized first, and what does this suggest about flow quality?

**Worked solution**

Priority should go first to Approval wait (22 min), then Verification wait (18 min), because they are the largest queues. This shows poor flow continuity in the middle of the process and directs the next observation toward the handoffs, release rules, and available capacity around those queues.

#### 17.4.4. Exercise 4

**4.** Analyze A team recorded processing times for the same process step over five runs (in minutes):

$$2,\ 3,\ 4,\ 9,\ 2$$

1. Which record most needs additional context in the event log?
2. What should the observer check or record before deciding how to interpret it?
3. Should the team treat these five records as a stable baseline yet?

**Worked solution**

1. The 9-minute record most needs additional context because it differs sharply from the other four.
2. Check the timestamps for a recording error and preserve factual context such as the work type, staffing, interruption, or equipment condition in Notes.
3. Not yet. Five runs are a practice starting point, and the unexplained 9-minute record is a reason to keep observing rather than discard it or declare the process stable.

#### 17.4.5. Exercise 5

**5.** Apply Select one recurring process from work or daily life and complete the following before drawing anything:

1. Write the start event, stop event, and unit of analysis.

2. Prepare event-log fields for the case ID, any child item ID needed when work splits, repeated passes, roles, timestamps, and factual notes.
3. Identify one likely queue point and one likely handoff risk to verify during observation.

**Worked solution**

Answers will vary. A usable response names observable start and stop events, follows one consistent kind of case, keeps the case ID when work splits into separately processed items, prepares row-level timestamp fields rather than prefilled averages, and treats the proposed queue and handoff as things to verify rather than established causes.

#### 17.4.6. Exercise 6

**6.** (Apply) Draft one observation note that is high quality and one that is low quality. Briefly explain why another reader can check the high-quality note.

**Worked solution**

**High quality:** "Order 14 entered the approval queue at 10:12 a.m.; work began at 10:31 a.m.; the approver said the prior meeting ran late."
**Low quality:** "Approval is always slow because the manager is disorganized."
The first note identifies the work unit, timestamps, and the source of the explanation while keeping observation separate from a participant's account. The second generalizes from no stated evidence and presents an assumed cause as fact.

#### 17.4.7. Exercise 7

**7.** (Analyze) An event log shows a long wait before approval but does not identify the roles on either side of the handoff. Explain what information is missing and why that weakens countermeasure design.

**Worked solution**

The log does not show which role releases the work, which role receives it, what authorizes the transfer, or when responsibility changes. Without that information, the team cannot tell whether the queue reflects a release rule, missing information, unavailable capacity, or some other mechanism, so a proposed countermeasure would outrun the evidence.

#### 17.4.8. Exercise 8

**8.** Understand Why should process variants be documented during observation even if only one primary baseline map is built first?

**Worked solution**

Variant records preserve which units followed different paths and why. That evidence prevents the primary map from being mistaken for a universal path and lets the team decide later whether a variant is necessary, wasteful, or important enough to map separately.

#### 17.4.9. Exercise 9

**9.** Analyze A team has five observations with one extreme outlier. Describe one decision rule for when to keep collecting versus when to use the observations as a provisional baseline.

**Worked solution**

Keep the outlier unless a documented measurement or data-entry error justifies correcting it, record its operating context, and collect more observations whenever the team cannot yet explain whether that context is recurring or exceptional. Five observations are enough for this practice exercise, but not enough to declare a stable baseline merely because four values are close together.

#### 17.4.10. Exercise 10

**10.** [Stretch] Evaluate You must choose between two first-cycle observation approaches:

1. observe one process variant deeply,
2. observe three variants lightly.

Which approach is better for a first White Belt cycle, and why?

> **Worked solution**
>
> Usually approach 1 is better. A single, well-observed baseline yields clearer metrics and actionable countermeasures, while three shallow maps often spread effort too thin and delay improvement. If variant risk is high, note variants during observation, but keep one primary baseline for the first cycle.

### 17.5. Building the Current-State Process Map

#### 17.5.1. Exercise 1

**1.** (Analyze) Using the completed Wash n' Fold table, explain why Wash, Dry, and Fold and package are VA while Drop-off and ticket, Sort and tag, and Stage and notify are RNVA. Identify one NVA queue and one NVA rework activity from the same evidence.

> **Worked solution**
>
> Wash, Dry, and Fold and package directly transform the laundry into the clean, dry, folded product the customer receives and values, so they are VA. The other three main activities support traceability, safe processing, staging, and communication without directly transforming the laundry, so they are RNVA. The Batch Release queue is NVA waiting, and Fold correction is NVA rework.

#### 17.5.2. Exercise 2

**2.** (Analyze) A small dental practice handles patient appointments as follows: the room is cleaned and prepared, the patient checks in at reception, waits in the waiting room, is called to the exam room, the hygienist performs the cleaning, the dentist reviews the x-rays, performs the exam, and the patient checks out and schedules the next appointment.

Using SPPM conventions, classify each step as VA, RNVA, or NVA and briefly justify each classification.

**Worked solution**

| Step | Category | Justification |
|---|---|---|
| Room setup and turnover | RNVA | Required preparation; record it as setup rather than queue wait. |
| Check in at reception | RNVA | Required for scheduling and billing; no value added to the patient's health. |
| Wait in waiting room | NVA | Pure queue; no value is added. |
| Called to exam room | RNVA | Necessary handoff, but no clinical transformation. |
| Hygienist cleaning | VA | Directly improves the patient's dental health. |
| Dentist reviews x-rays | VA | Clinical analysis that informs treatment. |
| Dentist exam | VA | Direct clinical value. |
| Check out and schedule | RNVA | Required for continuity and billing. |

### 17.5.3. Exercise 3

**3.** (Apply) Write a one-paragraph evidence summary of the representative Wash n' Fold Friday supplied by the simulation. Include the boundary, unit of analysis, largest queue, one rework observation, and one investigation target.

**Worked solution**

A strong response follows one customer order from drop-off through staging and readiness notification, identifies the 501.7-minute order-level mean Batch Release wait as the largest queue, notes that five Fold corrections were recorded among 39 orders, and proposes investigating the serial batch-release rule. It stays descriptive and does not claim a root cause that the observation has not yet tested.

### 17.5.4. Exercise 4

**4.** Evaluate A teammate proposes skipping direct observation and building the map from interviews to save time. Decide whether this is acceptable for a White Belt baseline, justify your decision, and give two evidence-quality risks.

**Worked solution**

Usually no. Interviews provide context, but observation is needed to capture actual sequence, queue behavior, load handling, and rework frequency. Memory and perception can distort timings, and hidden waits or workarounds are rarely reported consistently without direct observation.

### 17.5.5. Exercise 5

**5.** Apply Point to where each artifact appears in the Wash n' Fold application and explain what it contributes:

(a) process boundary statement,
(b) learner event log,
(c) completed activity summary,
(d) verified current-state map.

Then explain why the chapter builds them in that order.

**Worked solution**

The boundary fixes the start, stop, and unit of analysis. The event log preserves row-level timestamps and context. The activity summary aggregates that evidence without discarding the source records. The verified SPPM makes the dominant queue, load behavior, and correction path visible. The order matters because each artifact constrains the next one and prevents the map from outrunning the evidence.

### 17.5.6. Exercise 6

**6.** Apply A small print shop handles rush poster orders with this observed flow:

1. Receive order details and file from customer (4 min)
2. Queue before design check (18 min)
3. Design check and print setup (10 min)
4. Queue before printer availability (25 min)
5. Print run (12 min)
6. Trim and package (6 min)
7. Queue before pickup notification (20 min)
8. Send pickup notification (3 min)

Apply the SPPM mindset:

(a) Name the process boundary.
(b) Identify the two biggest queues.
(c) Calculate a first-pass total lead-time estimate.
(d) Suggest one practical target for the first PDCA cycle.

**Worked solution**

**(a)** Start when the customer submits the order and end when the customer is notified that it is ready.
**(b)** The printer queue (25 min) and pickup-notification queue (20 min) are largest.
**(c)** Total processing and setup time is 35 minutes and queue time is 63 minutes, so estimated lead time is **98 minutes**.
**(d)** Investigate the 25-minute printer queue first because it is the largest observed delay.

#### 17.5.7. Exercise 7

**7.** Apply For the print-shop flow in Q6, classify each queue as likely release-control delay or likely capacity-constraint delay. State one observation you would collect next to validate each classification.

**Worked solution**

- **Before design check (18 min):** provisionally a handoff or release-control delay. Record when completed files become visible to the designer, what authorizes the check, and whether the designer is available while work waits.
- **Before printer availability (25 min):** provisionally a capacity-constraint delay. Record printer occupancy, downtime, job mix, and queue length through the observation period.
- **Before pickup notification (20 min):** provisionally a release-control delay. Compare package-completion and notification timestamps and observe whether notifications are sent immediately or accumulated.

These are hypotheses to test; the durations alone do not establish their causes.

### 17.5.8. Exercise 8

**8.** (Analyze) A team map shows no rework path, but three orders were relabeled in one shift. Explain why this is a mapping defect and identify how the rework should be represented and quantified.

**Worked solution**

The map hides observed work that returned to an earlier activity, so it understates both effort and delay. Add a dashed rework path from the point where the labeling defect was detected back to the labeling activity, preserve each correction as its own activity record in the event log, and label the path as three recorded corrections with the relevant denominator, such as three affected orders out of the orders observed during that shift.

### 17.5.9. Exercise 9

**9.** (Evaluate) Review this draft: "Each customer order uses one washer and one dryer; there is no correction path; Stop means pickup and payment." The operator says, "Some orders need compatible loads that reunite before folding. Keep pickup outside our improvement boundary. I also think breakdowns must be a major problem." The log records 26 two-load orders and five corrections among 39 orders, with no recorded breakdowns. Produce a

revision record with proposed change, accepted/rejected/not yet supported, supporting evidence, and one follow-up question.

**Worked solution**

| Proposed change | Decision | Evidence | Follow-up question |
|---|---|---|---|
| Show compatible loads that reunite before folding. | Accepted | The log records 26 two-load orders, consistent with the operator's explanation. | Which order attributes trigger each split? |
| Add a Fold correction path. | Accepted | Five corrections are recorded among 39 orders. | What defect triggered each correction? |
| End the map at staged-and-notified rather than pickup and payment. | Accepted | The approved process boundary excludes customer travel, pickup, and payment. | Does any readiness notification occur before staging is actually complete? |
| Add a machine-breakdown path. | Not yet supported | The operator suspects breakdowns, but the selected log records none. | Which maintenance or downtime record could test this claim across more Fridays? |

### 17.5.10. Exercise 10

**10.** (Analyze) Compare a Wash n' Fold first-pass draft that treats every order as one load with the verified map. Identify two differences and explain why they matter for improvement decisions.

**Worked solution**

The verified map shows that many orders split into compatible loads and must reunite before folding; the first-pass map hides that coordination and would distort capacity and handoff reasoning. The verified map also includes the evidenced Fold correction branch and its frequency; omitting it would understate non-value-added work and remove a quality signal from the countermeasure discussion.

#### 17.5.11. Exercise 11

**11.** (Apply) Record or construct five event-log rows for one activity in a process of your choice. Calculate P/T, W/T, and S/T from displayed timestamps, then calculate the count, sum, and mean for P/T. State which values belong on the Map Summary Sheet, which value belongs on the map, and what warning should accompany it.

**Worked solution**

Answers will vary, but every row should use the same activity and time unit, preserve Queue Entry, Setup Start, Work Start, and Work End, and calculate nonnegative P/T, W/T, and S/T consistently. The Map Summary Sheet should preserve $n = 5$, the P/T sum, the mean, and a reference to the supporting log. The map should display **Avg P/T** with $n = 5$. The warning should explain that five observations are practice evidence rather than proof of stability and that the mean may conceal variation and relevant operating context.

### 17.6. Measuring and Interpreting Process Performance

#### 17.6.1. Exercise 1

**1.** (Apply) A registrar's office has ten eight-hour working days to handle 400 requests. Calculate Takt in minutes per request. One dedicated worker needs 11 minutes per request on average. What does that suggest, and what assumptions should you check?

**Worked solution**

Available time is $10 \times 8 \times 60 = 4,800$ minutes; Takt is $4,800/400 = 12$ minutes per request. Eleven minutes suggests that one continuously available worker could meet average demand under this simplified model. Check interruptions, rework, arrival bursts, other duties, and the actual staffed time.

### 17.6.2. Exercise 2

**2.** Apply A mortgage process has 15 minutes of RNVA intake, two workdays waiting, 45 minutes of VA review, three workdays waiting, 90 minutes of VA underwriting, and 10 minutes of RNVA notification. Use eight-hour workdays. Calculate lead time and PCE. If VA time stays unchanged, what maximum lead time corresponds to a 20% PCE target?

**Worked solution**

Lead time is $15+960+45+1,440+90+10 = 2,560$ minutes. VA time is 135 minutes, so PCE is $135/2,560 \times 100\% \approx 5.3\%$. The target lead time is $135/0.20 = 675$ minutes. The target is an exercise assumption, not an industry benchmark.

### 17.6.3. Exercise 3

**3.** Apply A repair shop averages 24 open repairs and completes six per working day. Use Little's Law to calculate average lead time. If throughput remains six per day, what average WIP corresponds to a three-day target? Explain why moving unfinished repairs into an uncounted storage area does not achieve the customer goal.

**Worked solution**

Average lead time is $24/6 = 4$ working days. The target corresponds to $6 \times 3 = 18$ average open repairs. Moving work changes the count's boundary without reducing customer time in the complete process.

### 17.6.4. Exercise 4

**4.** Apply Five activity P/T observations are 4, 6, 6, 9, and 15 minutes. Calculate count, sum, minimum, maximum, range, mean, median, mode, and

the nearest-rank five-number summary. A second group contains 15 observations with a sum of 210 minutes. Calculate the combined mean.

**Worked solution**

Count = 5; sum = 40 min; minimum = 4; maximum = 15; range = 11; mean = 8; median = 6; mode = 6 minutes. The nearest-rank five-number summary is 4, 6, 6, 9, and 15 minutes. The combined mean is $(40+210)/(5+15) = 12.5$ minutes. Averaging the two group means, 8 and 14, would incorrectly give 11.

#### 17.6.5. Exercise 5

**5.** Apply Reconstruct WNF-003's order row from its event-log records. Show the arrival-to-notification calculation and the P/T + W/T + S/T check. Count its activity records, named activities, correction records, and distinct affected orders. Explain why a worksheet row for Fold correction uses a different denominator from mean correction time per order.

**Worked solution**

$178.4 - 12.1 = 166.3$ minutes; $111.3 + 50.8 + 4.2 = 166.3$ minutes. There are nine activity records representing seven named activities, including one correction record, and one affected order. Across the Friday, correction averages use five correction records; its contribution per order uses all 39 orders.

#### 17.6.6. Exercise 6

**6.** Analyze Inspect Figure 9.1. Explain why the matching averages do not justify treating its four distributions as equivalent. Then compare these two ten-order cohorts, whose lead times are in minutes:

A: 10, 10, 10, 10, 10, 10, 10, 10, 10, 10.

B: 5, 5, 5, 5, 5, 5, 5, 5, 25, 35.

All have the same promise of completion within 20 minutes of arrival. Compare mean, median, nearest-rank P90, and promise-violation rate. What does the shared mean conceal?

**Worked solution**

Both means are 10 minutes. A has median 10, P90 10, and 0% violations. B has median 5, P90 25, and 20% violations. The shared mean conceals different experiences in the middle and slow tail. Anscombe's quartet demonstrates the broader point: matching summaries can coexist with a linear pattern, a curve, an unusual observation, or a result dominated by one distant point.

### 17.6.7. Exercise 7

**7.** Analyze Using the 21-row Wash n' Fold table, verify the mean, median, and nearest-rank P90. Then use Figure 9.2 and Figure 9.4 to describe one pattern that is difficult to see in the table alone. Check eligibility and count promise violations. Explain why WNF-016 is late despite a lead time shorter than 600 minutes.

**Worked solution**

Mean = $8,899.2/21 \approx 423.8$ min; median = position 11 = 424.5; P90 = position 19 = 676.2. All arrivals are at or before minute 120; six ready timestamps exceed 600. Violation rate = $6/21 \approx 28.6\%$. WNF-016's ready timestamp is 686.3; the promise is measured against the clock, not a 600-minute allowance from each arrival. The plots also reveal seven clusters of roughly three orders, widening lead times across those groups, and a slow tail containing the promise misses.

### 17.6.8. Exercise 8

**8.** Analyze The Wash n' Fold Batch Release mean wait is 501.7 minutes. The worksheet includes the same waiting interval that the map promotes to a queue triangle. Explain why adding both would be wrong. Name one supported symptom, one release-policy hypothesis, and the next observation needed.

**Worked solution**

The worksheet value and queue triangle are two representations of the same pre-Wash waiting interval, so adding them would count that delay twice. The supported symptom is a 501.7-minute mean wait before Wash. A release-policy hypothesis is that the serial batch rule holds orders even while washer capacity is available. The next observation should compare queued orders, the rule's release decisions, and idle washer positions over time.

#### 17.6.9. Exercise 9

**9.** Analyze A team removes intake and the release queue from its reported lead time but retains the full process's VA numerator. Its PCE rises. Explain the boundary error and describe a fair comparison.

**Worked solution**

The team shortened the PCE denominator by excluding elapsed time inside the original customer-facing process while leaving the numerator drawn from that larger process. A fair comparison uses the same start and stop events, cohort, clock, and VA classification before and after the change, or explicitly recalculates both numerator and denominator for a genuinely different declared boundary. Moving delay outside the report does not improve the customer's experience.

#### 17.6.10. Exercise 10

**10.** Apply A stable application service averages 12 open applications and an end-to-end lead time of three working days. Estimate completions per day, choosing the appropriate relationship yourself. Could a closing snapshot of 12 applications replace the measured average? Explain.

**Worked solution**

Throughput = average WIP / mean lead time = $12/3 = 4$ applications per working day. A closing snapshot does not establish average WIP; the count may change during the period.

### 17.6.11. Exercise 11

**11.** Analyze A colleague adds the Wash median, Dry median, and Fold median and calls the result median process lead time. Identify two problems. Explain which records would support the correct calculation.

**Worked solution**

Activity medians need not add to the median of order totals. The calculation also omits other activities, waits, setup, and any correction. Build complete order lead times from matching start/stop timestamps, then sort those totals to find their median.

### 17.6.12. Exercise 12

**12.** Analyze Write a capstone-ready statement using the representative Friday. Name the cohort, boundary, one metric symptom, one waste hypothesis, and the next evidence needed. Keep simulated observations, planning estimates, and causal claims distinct.

**Worked solution**

A strong statement should resemble the following:

> In the simulated representative Friday, the 39 orders followed from drop-off through staging and readiness notification had a mean operating lead time of 662.6 minutes, including a 501.7-minute mean wait before Wash. This supports investigating waiting associated with the serial batch-release rule, but it does not by itself prove that the rule caused the full delay. The next check should record queued orders, release decisions, and available washer positions across comparable Fridays before the team selects a countermeasure.

The statement should identify the observations as simulated, keep the 39-order cohort distinct from the 21 promise-eligible orders, and avoid presenting the 3.56-orders-per-hour capacity estimate as observed throughput.

## 17.7. The Eight Wastes

### 17.7.1. Exercise 1

**1.** Understand Define four of the Eight Wastes in your own words and give one realistic example of each from a service process.

> **Worked solution**
>
> Answers vary. A strong response uses accurate definitions and examples that match the waste category clearly (for example, waiting as idle queue time, defects as rework-triggering errors).

### 17.7.2. Exercise 2

**2.** Analyze In Wash n' Fold, mean release wait is 501.7 minutes; five Fold corrections were recorded among 39 orders; six of 21 eligible orders miss the promise. Complete a Waste Diagnosis Brief: identify two evidenced wastes, one supported explanation, one competing explanation, and the next observation that would help distinguish them. Explain why these observations do not prove overproduction.

> **Worked solution**
>
> Waiting and defect-related correction are evidenced. The serial release rule is a supported explanation: it may leave usable capacity idle while orders wait before Wash. Insufficient effective capacity is a competing explanation. Observe release decisions, queued orders, and resource occupancy under comparable demand to distinguish them. A queue alone does not show that upstream production outruns downstream capacity, and the correction records do not establish incorrect tags or mixed orders.

### 17.7.3. Exercise 3

**3.** Analyze A team proposes buying faster equipment before mapping or measuring waste. Analyze this proposal using the least-cost-way principle.

**Worked solution**

The proposal is usually weak without prior waste analysis. If most lead time loss is from waiting, transport, or rework, faster equipment may add little value versus lower-cost flow and standardization fixes. Map and baseline first, then justify capital only if the bottleneck remains equipment-limited.

### 17.7.4. Exercise 4

**4.** (Understand) Define each of these waste pairs in one sentence each and state one practical difference:

1. Waiting vs. Inventory,
2. Motion vs. Transportation,
3. Defects vs. Overprocessing.

**Worked solution**

1. **Waiting** is idle time before the next needed action; **Inventory** is unfinished work or material held in the system. A queued claim can be both, but the terms describe its delay and its accumulated presence, respectively.
2. **Motion** is unnecessary movement by a person within the work; **Transportation** is unnecessary movement of the work, customer, information, or material between places.
3. **Defects** create incorrect or incomplete output that requires correction; **Overprocessing** adds work beyond what the customer or process requires even when the output is not defective.

### 17.7.5. Exercise 5

**5.** (Analyze) In a claims process, analysts repeatedly request missing fields from customers. Classify the primary waste and one likely secondary waste, then justify both choices.

**Worked solution**

The primary waste is **Defects** because incomplete submissions require correction. **Waiting** is a likely secondary waste while the claim sits for the missing information; repeated checking or duplicate entry could also create **Overprocessing** if the observation supports it. A strong answer separates what the scenario establishes from what would still need observation.

#### 17.7.6. Exercise 6

**6.** Analyze A manager says, "If everyone stays busy, waste is low." Analyze why this claim can be misleading in a queued system.

**Worked solution**

Local activity can remain high while work accumulates between steps, customers wait, and downstream capacity is overwhelmed. Releasing more work to keep one role busy may increase WIP and lead time without increasing completed output. Judge the system with flow, quality, and customer measures, not utilization alone.

#### 17.7.7. Exercise 7

**7.** Understand Explain why waste classification must be tied to customer value and process boundary, not only local team effort.

**Worked solution**

An activity can look productive to one team while adding nothing the customer values or merely shifting delay outside that team's boundary. Naming the customer, unit, start, and stop keeps the classification attached to the end-to-end result and prevents local effort from being mistaken for value.

#### 17.7.8. Exercise 8

**8.** Apply For one service process you know, list two likely "invisible" wastes that customers feel but teams often normalize.

**Worked solution**

Answers vary. For each waste, name a specific customer-visible effect, an observable process sign, and the waste category. Examples include a request waiting unacknowledged in a queue, repeated transfer of the same information, or correction caused by an ambiguous intake field. Do not infer a cause or fill an unused waste category merely to complete the list.

#### 17.7.9. Exercise 9

**9.** Analyze The Wash n' Fold team could place supplies at each washer or change the serial release rule. Which observed or suspected waste would each change address, and which seems more likely to improve the 5:00 p.m. customer promise? Name the evidence you would check before choosing a test.

**Worked solution**

Point-of-use supplies would address setup-related motion and searching. Changing the serial release rule would address the much larger observed release wait and is therefore more likely to improve the customer promise. Before testing, check supply-retrieval time, order-release decisions, queue length, resource occupancy, demand, promise eligibility, and safety or workload effects. This evidence supports a bounded comparison; it does not prove the release rule is the sole cause of lateness.

### 17.8. Designing the Improved Workplace: Flow, Pull, and 5S

#### 17.8.1. Exercise 1

**1.** Understand In your own words, explain the difference between **Sort** and **Set in Order**. Give one practical example of each from either a physical or digital workspace.

**Worked solution**

Sort decides what should be present; Set in Order decides where and how needed items should be arranged for fast, reliable access. Strong examples clearly separate disposal/reduction decisions (Sort) from placement/layout decisions (Set in Order).

### 17.8.2. Exercise 2

**2.** Apply Complete a Wash n' Fold Trial Proposal using the fields modeled above. Identify the primary operating change, any supporting flow or 5S changes, mechanism, owner, local and customer measures, guardrails, and retain/revise/abandon criteria. State which parts require standard work and why.

**Worked solution**

Answers vary. A strong response makes release control the primary change because it addresses the dominant measured delay, while treating prompt notification and point-of-use supplies as supporting changes with their own local measures. It keeps all open orders visible, names correction and safety guardrails, and does not use an end-to-end package result as proof that every component worked. Standard work is appropriate where the proposal creates a repeatable release, notification, replenishment, or setup method.

### 17.8.3. Exercise 3

**3.** Apply A team reports that after a cleanup day, the area looked great for one week, then gradually returned to clutter. Which parts of 5S are most likely weak in their approach, and what evidence supports your diagnosis?

**Worked solution**

Standardize and Sustain are most likely weak. Short-lived gains suggest there was no stable standard work or routine audit/ownership system to maintain conditions after the initial event.

### 17.8.4. Exercise 4

**4.** (Evaluate) Evaluate this statement: "5S is mostly about neatness, not process performance." Do you agree or disagree? Defend your answer with one operations argument and one customer-impact argument.

**Worked solution**

Disagree. 5S can change how work is performed: for example, placing needed supplies within reach can reduce retrieval time. That may help orders finish sooner, but the customer benefit needs checking against the other delays in the process. A persuasive answer explains both the local mechanism and the customer outcome it could affect.

### 17.8.5. Exercise 5

**5.** (Understand) Define each 5S element in one sentence and name one common implementation mistake for each.

**Worked solution**

- **Sort:** retain what the work needs and remove or relocate what it does not; a common mistake is discarding items before asking the people who use them.
- **Set in Order:** arrange needed items so their location and use are clear; a common mistake is optimizing appearance instead of access and flow.
- **Shine:** clean while inspecting for abnormal conditions; a common mistake is treating cleaning as cosmetic labor with no inspection purpose.
- **Standardize:** define the repeatable condition and method; a common mistake is writing a rule without an owner, trigger, or abnormal-condition response.
- **Sustain:** review, learn from, and maintain the method; a common mistake is relying on memory or enthusiasm after the initial event.

### 17.8.6. Exercise 6

**6.** Apply Pick one workstation and draft a 5-day 5S trial plan with clear ownership for each day. Address safety throughout and include a point at which observations could lead you to revisit an earlier decision.

**Worked solution**

A usable plan names the workstation and owner; assigns observable Sort, Set in Order, Shine, Standardize, and Sustain work across the five days; protects access, ergonomics, and other relevant safety conditions; and states what will be measured before and during the trial. It also includes a review point at which evidence can reverse an earlier placement or removal decision rather than treating the sequence as irreversible.

### 17.8.7. Exercise 7

**7.** Apply Write a simple sustain audit checklist with five yes/no checks and one weekly review question. Name its owner and an escalation trigger for a high release queue that persists despite compliance with the workplace standard.

**Worked solution**

The five checks should describe visible conditions that two reviewers could judge consistently, not vague qualities such as “organized.” A complete response names an owner, asks whether the standard still helps safe and reliable work, and escalates a persistent release queue for flow or capacity investigation instead of prescribing more cleanup.

### 17.8.8. Exercise 8

**8.** Understand Explain why Safety is treated as integrated with 5S in this chapter instead of a separate afterthought.

**Worked solution**

Every 5S decision can change reach, lifting, travel, visibility, storage, access, or exposure to a hazard. Checking safety during each step allows the team to reject an unsafe arrangement before it becomes the standard rather than auditing harm after the design is complete.

#### 17.8.9. Exercise 9

**9.** Apply If your proposal creates or changes a repeatable operating method, complete the one-page standard work template for that method. Include its purpose, boundary, trigger, owner, sequence, quality and safety checks, abnormal conditions, escalation route, measures, and review cadence. If it does not require standard work, explain why an owner and review check are sufficient.

**Worked solution**

Answers vary. The method should be executable by another trained person and include every field named in the prompt. Its measures should test both the local mechanism and the customer outcome, while abnormal conditions and escalation rules should prevent staff from improvising silently when the method does not fit. A no-standard-work answer is acceptable only when the proposal does not create a repeatable sequence and still assigns ownership and review.

#### 17.8.10. Exercise 10

**10.** Apply Select a digital workspace (shared drive, ticket board, or repo). Apply 5S language to it and list one action for each S.

**Worked solution**

Answers vary. A strong response removes or archives unneeded material for Sort, gives active material an unambiguous location for Set in Order, checks and repairs stale or broken content for Shine, defines a naming or workflow rule for Standardize, and assigns a review cadence and owner for Sustain. The actions should improve retrieval or reliability, not merely make the workspace look tidy.

### 17.8.11. Exercise 11

**11.** Apply A claims team has three dedicated operators, 480 available minutes, and demand for 40 cases. Its sequential work times are 6, 14, and 8 minutes per case. Calculate Takt and explain the capacity mismatch. Describe one conditional rebalancing option, then write a trial pull rule with a limit, release signal, signal-return event, and treatment of blocked work. Name one measure that would warn you the limit was too low.

**Worked solution**

Takt is 12 minutes per case; the dedicated 14-minute stage cannot sustain that average pace. If three minutes of its work can legitimately move to the last stage, the loads become 6, 11, and 11; check skills, sequence, safety, and quality before trying it. This changes work allocation, not the customer's demand or the number of workers.

A pull proposal needs an explicit controlled boundary and an initial limit justified by observed work and capacity; the three processing times alone do not determine the right WIP ceiling. For a stated trial of four cases from review start through completion, a completion returns one card and authorizes the next ready FIFO case. A blocked case retains its card and remains visible; when all four cards are occupied, investigate the blockage rather than secretly adding a fifth. Monitor productive capacity left idle while ready customer work waits, along with throughput, end-to-end lead time, quality, and promise performance. Neither a lower WIP count nor everyone staying busy is sufficient evidence of improvement.

### 17.8.12. Exercise 12

**12.** Apply State what you will measure to check your flow, pull, and 5S trial, and what result would make you retain, revise, or abandon the proposed change.

**Worked solution**

Name the proposed operating change, measures comparable with the baseline, and the result that would change your recommendation. It should not announce success before the trial evidence exists.

## 17.9. Check and Act: What Did We Actually Improve?

### 17.9.1. Exercise 1

**1.** (Evaluate) Suppose your pilot shows improved lead time but a slight rise in defect/rework rate. Would you standardize now, revise and re-test, or roll back? Defend your decision.

**Worked solution**

Most strong responses choose revise and re-test unless defect risk is trivial and controlled. White Belt judgment should balance flow gains against quality stability before standardization.

### 17.9.2. Exercise 2

**2.** (Apply) Write a one-paragraph Act-phase standardization note for one improvement that worked in your case. Include owner, trigger, review cadence, and the condition that would reopen the method for revision.

**Worked solution**

A good note names who owns the new standard, when it is triggered, how often compliance and performance are reviewed, and what result requires reconsideration. It should be specific enough that another team member can execute it without ambiguity.

### 17.9.3. Exercise 3

**3.** (Apply) Draft the opening Plan statement for the next PDCA cycle based on what remained unresolved. Include one problem statement sentence and one SMART goal sentence.

**Worked solution**

Answers vary. A strong response is concrete, measurable, and explicitly connected to observed results from the prior cycle.

### 17.9.4. Exercise 4

**4.** (Evaluate) Your after-state improves lead time but leaves rework unchanged. Would you accept, revise, or reject the change? Defend your decision using two criteria.

**Worked solution**

The evidence can justify retaining the lead-time change while revising the next cycle to address rework; it does not justify claiming that the whole problem is solved. A strong response uses at least one customer or flow criterion and one quality or risk criterion, checks any predeclared guardrails, and explains why the chosen Act decision fits both. Reject or roll back the change if the unchanged rework violates an acceptance rule or leaves an unacceptable harm; otherwise preserve the supported gain and keep the correction gap visible.

### 17.9.5. Exercise 5

**5.** (Analyze) Audit the software-produced 21-order comparison. Confirm that its cohort, boundary, clock, physical unit, percentile convention, and promise denominator match the baseline. Verify the after-state violation count directly from the ready timestamps, then write a short Check summary that also notes the supplied all-order PCE result and unchanged correction count.

**Worked solution**

The comparison uses the same 21 eligible orders, drop-off-to-ready boundary, operating-minute clock, minutes-per-order unit, nearest-rank percentile convention, and 5:00 p.m. deadline as the baseline. All 21 after-state ready timestamps are at or before minute 600, confirming the software's zero violations, compared with six of 21 before. Mean operating lead time falls from 423.8 to 243.6 minutes, and P90 falls from 676.2 to 315.4 minutes. The zero violation result applies to this matched Friday, not every future Friday.
Across all 39 orders, the supplied PCE rises from 13.9% to 29.2%, while correction remains five occurrences. The package improves modeled flow and promise performance but does not remove the correction problem. Do not mix the all-order mean of 314.3 with the eligible-order mean of 243.6, or use all 39 orders as the promise denominator.

### 17.9.6. Exercise 6

**6.** (Evaluate) Before opening the answer, make a provisional Check/Act recommendation from this Friday. What would you retain for another test, what would you monitor, and what can you not yet attribute to 5S? Then inspect the calendar-year simulation evidence in the answer and explain whether it changes your recommendation.

**Worked solution**

A defensible provisional recommendation is to retain the package for further testing, monitor promise violations and all open orders, and continue investigating correction. Do not attribute the whole improvement to 5S: release, notification, and setup changed together. An alternative proposal is not wrong merely because it differs from this package; it needs comparable evidence.

The simulation also follows 52 calendar Friday demand streams through every scenario. Each matched set uses the same arrivals and underlying work requirements; the scenarios differ only in their specified operating rules. The Friday arrival intensity follows an explicitly modeled annual cycle ranging from 15% below to 15% above the typical level. This is an illustrative assumption for testing the methods under quieter and busier periods, not a claim about the seasonal pattern of an actual laundromat. The table reports averages of the 52 Friday-level measures, not one pooled set of orders.

**Table 17.1.:** Matched calendar-year answer evidence; operating L/T and daily P90 use all orders, while violations use each Friday's eligible cohort.

| Scenario | Mean operating L/T (min) | Mean daily P90 (min) | Mean daily miss rate | Mean setup/order (min) |
|---|---|---|---|---|
| Current serial batches | 657.6 | 1,053.5 | 27.60% | 4.23 |
| Pull only; batched notification; shared jug | 365.0 | 494.6 | 1.66% | 4.23 |
| Pull + prompt notification; shared jug | 339.8 | 468.8 | 0.43% | 4.23 |
| Pull + prompt notification + point-of-use supplies | 324.2 | 452.4 | 0.31% | 2.33 |

Release control accounts for the largest incremental reduction in mean operating L/T: approximately 292.6 minutes from current batches to pull only. Prompt notification adds about 25.2 minutes; point-of-use setup adds about 15.6 beyond that. These are differences between scenario means in this sequence, not independent effects guaranteed under every combination. There is no 5S-only scenario here, and no basis for claiming that setup changes alone would reproduce the combined gain.

Pull-only mean operating L/T is lower than current-state mean operating L/T on all 52 matched Fridays. That consistency supports the modeled release explanation; it is not independent validation of the model or a forecast for a real shop. The final scenario still has five late eligible orders across the simulated year, so the zero failures on Friday 91 should not be generalized.

Two aggregation details matter. First, these technical exports calculate each day's P90 by interpolation, then average those daily percentiles; they are neither the pooled P90 nor the nearest-rank P90 reported for the matched Friday in Exercise 5. Second, the mean daily violation rates weight Fridays equally. Pooling orders gives 314/1,007 ≈ 31.18% before and 5/1,007 ≈ 0.50% after, because Fridays have different eligible-order counts. Both summaries are valid when labeled; they answer different questions.

The stronger Act recommendation is to carry the release change forward with supporting notification and setup standards, then test whether an actual shop can implement them safely and reliably. Keep a named owner, a defined review window, a promise denominator, and an escalation trigger. Monitor correction separately: the simulation held correction decisions fixed, so improved flow does not demonstrate improved defect prevention.

#### 17.9.7. Exercise 7

**7.** (Apply) Prepare a six-bullet owner brief for your proposed improvement using the categories above. End with one difficult question you would want the owner or a process participant to ask before authorizing the trial.

**Worked solution**

Answers vary. A strong brief requests a bounded decision, connects the problem to customer and business consequences, explains the proposed mechanism, names costs and human burdens, distinguishes observed or simulated evidence from unknowns, and defines ownership and a review rule. The closing question should expose a real vulnerability—for example, whether the work limit fits large orders, whether another measure could worsen, or whether the people performing the work can carry out the method safely.

## 17.10. White Belt Capstone: Run the Cycle

### 17.10.1. Exercise 1

**1.** Analyze Review your capstone artifact set. Identify its weakest evidence link and one assumption in your recommendation that could fail in practice. State what observation or test would strengthen that link.

**Worked solution**

Name one specific link between the customer problem, observation, diagnosis, intervention, result, and recommendation rather than describing the packet generally. Identify the assumption separately, and propose an observation or bounded test that could reveal whether it is wrong.

### 17.10.2. Exercise 2

**2.** Evaluate Evaluate your metric set for blind spots. Name one important customer or process outcome currently unmeasured and explain how you would measure it next cycle.

**Worked solution**

The missing outcome should matter to the named customer or to a guardrail such as quality, safety, burden, cost, or timeliness. Define the unit, observation method, and denominator or comparison so another person could collect the measure in the next cycle.

### 17.10.3. Exercise 3

**3.** Apply Write a two-sentence executive summary for your capstone that includes the customer impact, evidence status, and next action.

**Worked solution**

The first sentence should state the customer problem and what the comparable evidence showed. The second should request a specific next action while preserving any important limitation or unresolved risk. Avoid claiming causation or lasting improvement beyond the evidence collected.

### 17.10.4. Exercise 4

**4.** Analyze Compare two recommendations: one with high impact and high effort, the other with moderate impact and low effort. Which would you stage first in Act, and what evidence supports that sequence?

**Worked solution**

Either sequence can be defensible. Compare expected customer benefit, strength of evidence, effort, reversibility, risk, and learning value; then state what result would authorize or stop the next stage. Do not choose on impact and effort labels alone.

### 17.10.5. Exercise 5

**5.** Evaluate A teammate argues that documenting what did not improve will weaken stakeholder confidence. Evaluate this claim and justify your response.

**Worked solution**

A strong response rejects hiding an unchanged or adverse result. Transparent limits make the recommendation more credible, protect customers and participants from unsupported rollout, and identify the next learning need. Explain how to communicate the supported gain and the unresolved gap together without exaggerating either one.

### 17.10.6. Exercise 6

**6.** Apply Identify a remaining problem supported by your capstone evidence and write a SMART goal for the next PDCA cycle.

**Worked solution**

Name an evidenced remaining gap and set a specific, measurable, achievable, relevant, and time-bound target for the next cycle. It should preserve the same unit and boundary unless the change is explicitly justified.

Part VI.

# Reference

# 18.

# Glossary

Terms are alphabetized. Follow a definition's explanation link to see it in context. Abbreviations and alternate names point to the same definition.

#### Batch Processing Production System

A batch-processing production system accumulates multiple units and processes them as a group before releasing the group to the next step.

Explanation in context.

#### Branching Decision Point

See Decision.

#### Business Functions

The functions are the things the firm has to do in order to operate, comprising broad areas such as HR, Sales, Finance, Operations, and potentially other functions such as legal or research and development.

#### Business process

See Process.

### C/O

See Changeover Time.

### C/T

See Cycle Time.

### Changeover Time

Changeover time, written **C/O**, is the setup time required to switch a process from one product, service, batch type, or operating mode to another. Every changeover is setup, but not every setup is a changeover. Use C/O only when the observation records an actual transition and identifies what changed.

Explanation in context.

### Common Cause Variation

Variation produced by the process as it currently operates: the combined effect of sources that are inherent in the system over time and affect its outcomes broadly.

### Continuous Flow Production System

In a continuous-flow production system, each unit begins the next step as soon as the preceding step is complete and the next step is available.

Explanation in context.

### Continuous Process Improvement

A philosophy that says our processes can always be improved.

### Customer

The customer of a firm is the person or organization that receives the benefit of its services or the use of its goods, regardless of who pays. The customer, payer, purchaser, and funder may be the same party, but they need not be.

Explanation in context.

### Customer Specification

The precise, measurable properties a finished good or service must have to meet a customer's need.

### Cycle Time

Cycle time, written **C/T**, is the observed time between successive completions from a process or process step. For repeating operator work, it can also describe the time required to complete the work elements before the sequence repeats. In a stable, sequential one-piece operation, one unit's P/T and the C/T between completions may be numerically equal; the labels still answer different questions. Cycle time is a completion pace, not the sum of processing times along a unit's path.

Explanation in context.

### Decision

A decision identifies the point where a process can follow alternative paths. If making the decision consumes work or waiting time, preserve that time in the log.

Explanation in context.

### Defect

A defect is a failure of a product or service to fulfill a stated requirement or a reasonable expectation for use, including applicable safety requirements.

Explanation in context.

### Distribution

The pattern of values observed for a quantity, including which values occur and how often or where they occur.

### Economy of Scale

A firm enjoys an economy of scale when average cost per unit falls as output increases over the relevant range.

### Event log

An event log is a row-by-row record of what happens to identified units of work as they move through a process. Each row preserves enough context to say what happened, to which unit, and when. The log is the evidence from which the map worksheet, process map, and order-level summaries are derived.

Explanation in context.

### Firm

A firm is an organized entity that coordinates people and resources to provide goods or services of value to customers.[1]

Explanation in context.

### First-Pass Yield

First-pass yield, or FPY, is the proportion of units that complete a defined step or process without a defect, correction, scrap, or rework on their first pass:

$$\text{FPY} = \frac{\text{Units completed correctly on the first pass}}{\text{Units entering the defined boundary}}$$

State the process boundary, observation period, and denominator. One unit can contain more than one defect, so FPY is not the same as a defect count.

Explanation in context.

[1]This book uses *firm* as an analytical term drawn from economics and organization theory for an organized unit that coordinates resources to produce goods or services. The usage is broader than ordinary commercial speech: when the same organizational questions arise in a public or nonprofit body, the book treats that body as a firm for purposes of analysis.

### Five-Number Summary

A compact distribution summary consisting of the minimum, first quartile, median, third quartile, and maximum.

### FPY

See First-Pass Yield.

### Hidden Factory

The unmeasured or poorly visible capacity consumed by rework, sorting, expediting, firefighting, workarounds, and other activities needed to compensate for failures in the official process.

### Industrial Engineering

An engineering discipline concerned with the creation of efficient production systems that integrate workers, materials, machinery, and information to produce products or provide services.

### Interruption

An interruption is a stop or diversion that pauses the normal flow of a work unit, usually to handle an urgent exception or priority task. Interruptions increase variation and make processing-time and completion-pace estimates less stable. Record the source, duration, and point in the activity in Notes so repeated patterns can be investigated later.

Explanation in context.

### L/T

See Lead Time.

### Lead Time

Lead time, written **L/T**, is the elapsed time between the defined start and stop events for one work unit. Unless another clock is stated, elapsed lead time includes all calendar time within that boundary.

Explanation in context.

### Lean Six Sigma

A methodology and body of knowledge that combines features of the Lean and Six Sigma traditions into a collection of tools used in carrying out process-improvement projects. The goal of Lean Six Sigma is to implement continuous improvement by eliminating waste and reducing variation.

### Least Cost Way

A feasible way of achieving the required result at the lowest relevant total cost among the alternatives considered. The comparison must include implementation, operating, quality, and safety consequences.

### Little's Law

The long-run relationship among work in process, throughput, and average lead time: $WIP = \text{Throughput} \times \text{Average Lead Time}$.

### Non-value-added process step

A non-value-added process step fails one or more parts of the value-added test and is not currently required to deliver the product or service safely and reliably.

Explanation in context.

### NVA

See Non-value-added process step.

### Operating Lead Time

Operating lead time counts only the periods in which the process is scheduled to operate between the defined start and stop events. Wash n' Fold uses drop-off to ready-and-notified and excludes closed hours. Report the operating calendar whenever this measure is used.

Explanation in context.

### Operations

The function of the business that produces the good or provides the service that the customer values.

### Operations research

A branch of applied mathematics concerned with optimization problems relevant to business operations, including logistics, supply chain management, and production scheduling.

### Operator Balance Chart

A chart that compares the work assigned to each operator with Takt. Bars above the Takt line identify assignments that cannot sustain the required pace under the stated assumptions; bars need not be equal for the process to meet Takt.

### P/T

See Processing Time.

### PCE

See Process Cycle Efficiency.

### PDCA

Plan–Do–Check–Act: define the problem and proposed test, carry out the test, compare results with expectations, and use what was learned to standardize, revise, or reject the change.

### Percentile

A value associated with a stated position in a sorted distribution; for example, P90 marks a value at or below which at least 90 percent of observations fall under the nearest-rank convention.

### Process

A business process is a set of steps designed to produce a specific product or provide a specific service to a customer.

Explanation in context.

### Process Cycle Efficiency

Process cycle efficiency, or PCE, is the share of lead time spent in value-added work:

$$PCE = \frac{\text{Value-added time}}{\text{Lead time}} \times 100\%$$

For a group of orders, this book's White Belt convention is to divide total VA time by total lead time for the same orders. That equals mean VA time divided by mean lead time for that group. It need not equal an unweighted average of individual orders' PCE percentages. When the study uses an operating clock, name the denominator as operating lead time and use that same clock throughout.

Explanation in context.

### Process Model

A representation of the relationships between the steps of a process and the resources those steps require, including personnel, information, materials,

machinery, and related elements. There are many kinds of process model, including flowcharts and Value Stream Maps.

#### Process run

A process run is one complete instance of a process, traced for one thing from the process's defined start event to its defined stop event. That thing might be one customer order, one patient visit, one service request, one batch of work, or one invoice.

Explanation in context.

#### Process Step

A process step is a discrete unit of work.

Explanation in context.

#### Process variant

A process variant is a recurring alternative path within the same process boundary. Variants share the same start event, stop event, and primary unit of analysis but differ in one or more steps, roles, timings, or decision paths.

Explanation in context.

#### Processing Time

**P/T** means processing time. For one pass through an activity, processing time is the elapsed time from Work Start to Work End while the unit is actively being processed. **Average processing time** is the arithmetic mean across comparable activity records. Write Avg P/T when transferring that mean to the worksheet or map.

Explanation in context.

### Production system

A production system is the organization of people, processes, machinery, materials, and information necessary to create a product or provide a service.

Explanation in context.

### Project Management Body of Knowledge

A collection of tools and techniques formulated by the Project Management Institute and used to train project managers in the administration of project-based work.

### Project Production System

A production arrangement organized around a particular, often one-off deliverable, with activities and resources coordinated to complete that project.

### Pull production

Release of work authorized by actual downstream need and available capacity, using an explicit signal and rule. A new arrival alone does not override a work limit.

### Push production

Release of work according to an upstream plan or forecast, without requiring an authorization signal from the receiving step.

### Queue

A queue is a state where work items sit idle, waiting for the next process step to begin. A queue adds wait time rather than processing time. Record all queue time in the event log's W/T calculation, record visible queue counts with their observation times, and draw a queue triangle when the waiting state is large or diagnostically important.

Explanation in context.

### Range

The observed maximum minus the observed minimum for a stated group of values.

### Red Tagging

Marking a questionable item during Sort so its use and disposition can be reviewed before it is retained, relocated, or removed.

### Redundant Steps

Steps that repeat work already completed without adding value or meeting a necessary requirement. On a process map, they are a reason to investigate why the repetition occurs before removing it.

### Required non-value-added process step

A required non-value-added process step fails one or more parts of the value-added test but is currently required by law, regulation, contract, a documented safety or control need, or another necessary condition of delivering the service reliably. Record the specific requirement rather than treating RNVA as a permanent exemption from improvement.

Explanation in context.

### Rework loop

A rework loop is a path where work returns to an earlier step because a defect, omission, or mismatch was found. Rework loops increase total effort and delay; they are a cost multiplier on total processing time and lead time and indicate quality problems. Add a row for each pass through the correction activity, linked to its work unit. On the worksheet, record the trigger, destination, and frequency with its denominator, such as 5 affected orders out of 39.

Explanation in context.

### RNVA

See Required non-value-added process step.

### Rolled Throughput Yield

Rolled throughput yield, or RTY, is the product of the first-pass yields for the defined stages of a process.

For a referral pathway with booking and completed-visit stages, define the cohorts explicitly:

$$\text{RTY} = \frac{\text{Referrals booked}}{\text{Referrals created}} \times \frac{\text{Booked referrals completed in network}}{\text{Referrals booked}}$$

When every completed in-network visit belongs to the booked cohort, the booked-referral counts cancel and RTY equals completed in-network visits divided by created referrals. The stage percentages should still be reported because the same RTY can conceal different points of loss.

### RTY

See Rolled Throughput Yield.

### S/T

See Setup Time.

### Setup Time

**S/T** means setup time. For one pass through an activity, setup time is the preparation interval from Setup Start to Work Start. **Average setup time** summarizes comparable observed preparation intervals; label it Avg S/T. Record it separately from queue wait time. Record `0` only when no setup occurs; leave unobserved setup time unknown.

Explanation in context.

### Special Cause Variation

Variation associated with an intermittent, unusual, or unpredictable influence that affects only some process outcomes. A statistical signal may indicate a special cause before the specific source has been identified.

### Swim-Lane Process Map

A process map organized into lanes for the roles, functions, teams, or systems that perform the work. The lanes make responsibility and handoffs visible; a swim-lane map is not, by that fact alone, a value-stream map.

### Takt Time

Takt time is available production time in a planning period divided by demand for that same period:

$$T = \frac{A}{D}$$

Explanation in context.

### Throughput

The average number of work units completed by a process per unit of time.

### Total Processing Time

Total processing time is the sum of the processing times on one unit's path, including repeated work. Average total processing time averages those unit totals. Active processing includes automatic machine cycles; it does not mean a worker is occupied throughout.

Explanation in context.

### Total Quality Management

A process-improvement methodology that preceded Lean Six Sigma.

### Toyota Production System

Toyota's production approach, centered on just-in-time flow and jidoka (detecting abnormalities and stopping to prevent defects from continuing downstream).

### Unit of analysis

The unit of analysis is the primary "thing" whose passage through the process counts as one run of the process. It might be one customer order, patient visit, service request, batch, or invoice. Choose it before recording observations so that process-level measures refer to comparable cases.

Explanation in context.

### VA

See Value-added process step.

### Value

The value of a product or service is its worth to the customer, judged from the customer's point of view.

Explanation in context.

### Value Stream Map

A Lean process model that represents the material or service flow and the information flow needed to bring a product or service from a defined request to delivery. A current-state map helps expose sources of waste and supports the design of a future state.

### Value-added process step

A process step is a value-added process step only when it passes this **three-part test**:

- **(i) Customer value:** The step directly creates or advances an outcome the customer receives and values.
- **(ii) Transformation:** The step changes the form, fit, function, information, or condition of the product or service toward that outcome.
- **(iii) Right first time:** The step is performed correctly without creating a need for correction or rework.

Explanation in context.

#### Variation

Differences among individual observations or outputs of a process. Variation may arise from common causes inherent in the current process or from special causes that affect only some outcomes.

#### W/T

See Wait Time.

#### Wait Time

**W/T** means wait time. For one pass through an activity, it is the elapsed time from queue entry until setup starts; if there is no setup, it ends when work starts. **Average wait time** is the arithmetic mean of those observed waits; label the summary Avg W/T. All queue time is wait time, whether or not you later choose to draw that queue as its own symbol on the map.

Explanation in context.

#### Waste

Consumption of effort, time, materials, or other resources that does not create customer value. Some non-value-added work is currently required; classification alone does not authorize its removal.

#### WIP

See Work in process.

**Work in process**

Unfinished work currently inside the defined process boundary.

# 19.

# Bibliography


Anscombe, F. J. 1973. “Graphs in Statistical Analysis.” *The American Statistician* 27 (1): 17–21.

Bryk, Anthony S., Louis M. Gomez, Alicia Grunow, and Paul G. LeMahieu. 2015. *Learning to Improve: How America’s Schools Can Get Better at Getting Better*. Harvard Education Press. https://www.education.pitt.edu/wp-content/uploads/2024/04/LearningToImprove_2015.pdf.

Flannery, Maura. 2007. “Observations on Biology.” *The American Biology Teacher* 69 (9): 561–64.

George, Michael L. 2002. *Lean Six Sigma: Combining Six Sigma Quality with Lean Production Speed*. 1st ed. McGraw-Hill.

Hennig, Alicia, and Edward Romar. 2023. “Learning from Japanese Businesses: Ethics in Operational Excellence.” *Humanistic Management Journal* 8 (3): 329–54. https://doi.org/10.1007/s41463-023-00160-w.

Hopp, Wallace J., and Mark L. Spearman. 1996. *Factory Physics*. Waveland Press.

Krafcik, John F. 1988. “Triump of the Lean Production System.” *Sloan Management Review* 30 (1): 41–52.

Liker, Jeffrey K. 2021. *The Toyota Way: 14 Management Principles from the World’s Greatest Manufacturer*. 2nd ed. McGraw Hill. https://www.mheducation.com/highered/mhp/product/toyota-way-second-edition-14-management-principles-world-s-greatest-manufacturer.html.

Ohno, Taiichi. 1988. *Toyota Production System: Beyond Large-Scale Production*. Productivity Press.

Rother, Mike, and John Shook. 1999. *Learning to See: Value-Stream Mapping to Add Value and Eliminate Muda*. Lean Enterprise Institute.

Tennant, Geoff. 2001. *Six Sigma: SPC and TQM in Manufacturing and Services*. Gower Publishing.

Womack, James P., and Daniel T. Jones. 1996. *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*. Free Press.