



*From the MixCache.com library*

SAMPLE COPY

# Systems That Scale

MixCache.com

SAMPLE COPY

## Table of Contents

- Introduction
- Chapter 1 — Diagnosing Your Business: Where Are the Bottlenecks?
- Chapter 2 — Design Thinking for Operations
- Chapter 3 — The Core Operating System: SOPs, Playbooks, and Routines
- Chapter 4 — Mapping Customer Journeys and Workflows
- Chapter 5 — Building a Reliable Sales System
- Chapter 6 — Marketing That Feeds Operations (not the other way around)
- Chapter 7 — Pricing, Packaging, and Unit Economics
- Chapter 8 — Hiring for Repeatability: Roles, Scorecards, and Onboarding
- Chapter 9 — Training Systems and Knowledge Transfer
- Chapter 10 — Delegation, Accountability, and Decision Rights
- Chapter 11 — Performance Management and Metrics that Matter
- Chapter 12 — Cash Flow, Forecasting, and Financial Systems
- Chapter 13 — Operations Technology Stack (Choosing Tools, Integrations, and Avoiding Overload)
- Chapter 14 — Automations, Templates, and Smart Repetitions
- Chapter 15 — Scaling Delivery: Capacity Planning and Workload Balancing
- Chapter 16 — Quality Assurance and Continuous Improvement
- Chapter 17 — Customer Success and Retention Systems
- Chapter 18 — Supply Chain, Inventory, and Fulfillment Basics (for product businesses)
- Chapter 19 — Legal, Compliance, and Risk Management for Small Teams
- Chapter 20 — Culture, Values, and Leadership Routines
- Chapter 21 — Growth Transitions: From Founder-Led to Manager-Led
- Chapter 22 — Partnerships, Channel Programs, and Strategic Growth Levers
- Chapter 23 — Franchising, Licensing, and Replicable Business Models
- Chapter 24 — Preparing for Exit or Next-Stage Funding
- Chapter 25 — The 90-Day Systems Sprint: A Playbook to Implement Fast

## Introduction

Most small businesses are built on heroics. A founder hustles, remembers everything, and plugs every gap. It works—until it doesn't. As sales rise, so do mistakes, delays, and late-night Slack messages. What once felt like freedom becomes a treadmill. This book exists to help you step off that treadmill by replacing heroics with systems: repeatable, measurable, documented ways of working that anyone on your team can run without you standing over their shoulder.

Consider two versions of the same Tuesday. In the hero-dependent business, the founder wakes to a text from a frustrated client, scans a cluttered inbox for a proposal she “knows she sent,” then spends the morning triaging tasks only she can do: pricing, approvals, firefighting. Lunch is a “working session” that bleeds into a late delivery and an apologetic email. The day ends with a half-finished forecast and a nagging sense that growth just means more chaos. In the systemized business, the founder opens a simple dashboard: green lights for cash, capacity, and pipeline; one yellow on customer onboarding. The team runs morning standups with clear agendas and KPIs. Sales hands off two closed deals using a standard checklist; delivery kicks off with a templated plan. The founder spends the afternoon on hiring interviews and a partner meeting. Both businesses are busy. Only one is scalable.

In this book, a “system” means three things working together: repeatable (the steps are clear and followed), measurable (inputs and outcomes are tracked in a small set of metrics), and documented (accessible SOPs, checklists, and templates that live where work happens). Systems don't eliminate judgment; they create room for it. By stabilizing the routine, they free you to focus on the exceptions, the customers who need nuance, and the strategic bets that move the company forward.

To ground this in reality, here are three short founder stories you'll meet throughout the book. First, Maya runs a five-person branding agency. Before systems, she personally reviewed every deliverable and wrote most proposals. Turnaround times slipped to 21 days and client churn reached 18% per quarter. After mapping her customer journey, introducing a standard scoping checklist, and handing proposal creation to a project manager using a templated library, her average turnaround dropped to 12 days, churn fell to 7%, and revenue per employee rose 28% in two quarters. She still reviews marquee work—but at defined gates, not in ad hoc midnight marathons.

Second, Luis co-founded a specialty e-commerce brand with lumpy demand. Stockouts were common, and he guessed at reorders. By implementing a simple 13-week cash and inventory model, defining reorder points, and creating a vendor scorecard with

SLAs, he cut stockouts by 60% and reduced cash tied up in slow movers by 22% in five months. The payoff wasn't just financial: Luis reclaimed his weekends, because the "what do we buy this week?" question became a spreadsheet-driven routine that his ops lead owned.

Third, Priya leads a ten-person B2B SaaS startup at the pre-Series A stage. Sales were founder-led, demos varied wildly, and handoffs to onboarding were painful. She introduced a qualification script, a standardized demo flow, and a service-level agreement between sales and customer success. Activation time fell from 18 to 9 days, NPS rose from 32 to 52, and Priya redirected 12 hours a week to hiring and partnerships. "We didn't get fancy," she told me. "We got consistent."

This is a practical playbook. You'll see familiar ideas—E-Myth's push toward working on the business, EOS/Traction's clarity around roles and rhythms, and Scaling Up's emphasis on focus and metrics—but translated into the smallest possible set of actions a busy team can execute this week. You won't find abstract theory for theory's sake. Each chapter opens with a real-world vignette, presents a concise framework, walks you through a step-by-step implementation plan, provides 2–4 templates or checklists, and ends with a short "Quick Actions" list you can complete in 30–60 minutes.

How to use this book: start by diagnosing where you are today (Chapter 1). Then choose one of three tracks based on your immediate bottleneck. If delivery is breaking, go to Chapters 3–5 and 15–17 to stabilize SOPs, handoffs, capacity, and customer success. If demand is the issue—but you don't want to outstrip capacity—use Chapters 5–7 and 11 to build a reliable sales system aligned with your unit economics and metrics. If leadership and clarity are missing, jump to Chapters 8–10 and 20–21 to fix hiring, delegation, and culture. If you're preparing for funding or exit, Chapters 12, 19, and 24 will help you present operational hygiene investors trust. Finally, Chapter 25 gives you a 90-day sprint to sequence the work and prove ROI quickly.

The artifacts are the point. You'll get an SOP template and naming convention so your documentation stays consistent, a hiring scorecard and interview script to raise the bar on talent, a sample KPI dashboard to make performance visible, a RACI matrix for decision rights, and lightweight tools like a CAC/LTV calculator, capacity planning spreadsheet, and renewal playbook. Use them as-is or adapt them; the goal is to make the default way of working the best way of working. Sidebars throughout the book include mini-templates and examples you can copy in minutes.

A note on scope and pacing: you don't need to implement everything at once. Systems compound when introduced in the right order. Stabilize the customer journey and handoffs before adding advanced automation. Define roles and scorecards before scaling hiring. Get a cadence of weekly and monthly reviews running before layering

on more KPIs. When in doubt, reduce the number of moving parts, shorten feedback loops, and make the work visible.

Throughout, you'll see short case vignettes from founders in agencies, local services, e-commerce, and B2B SaaS, plus quantitative snapshots that show what changed—revenue per employee, onboarding time, or hours freed for the founder. These are not unicorn stories; they are achievable with a focused 6-12 weeks of effort. Expect honest tradeoffs, too. Systems may slow you down for a few days as you document and train; the payoff is weeks and months of regained time and smoother growth.

By the end of this book, you should be able to answer five questions with clarity: What outcomes matter most to our customers? What are the steps—owned by whom—that produce those outcomes? How do we measure and manage the work weekly and monthly? Where do we invest next to relieve bottlenecks? And how do we grow without making the founder the bottleneck again? If you can answer those, you have a business that can scale steadily and run without you.

Let's begin by diagnosing where the work really happens, where it stalls, and how to make progress visible. Once you can see the system, you can improve it. And once you improve it, you can repeat it—at first with five people, then twenty-five, then fifty—without sacrificing quality, sanity, or the reason you started the business in the first place.

## CHAPTER ONE: Diagnosing Your Business: Where Are the Bottlenecks?

Marco runs a small construction consultancy. His firm designs high-end kitchen remodels and manages the build for affluent homeowners. Last year, they did \$1.8 million in revenue with nine employees, which sounds great until you dig into the week-to-week reality. On Monday mornings, Marco's inbox looks like a slot machine: a dozen emails from clients asking for updates, a handful from subcontractors demanding decisions, and a few from his project managers wondering where the revised drawings are. He spends his days jumping between scopes, pricing change orders, and placating clients who are frustrated about delays. He's working seven days a week, but the firm's net margin is stuck at 8%. The business is growing, but the only person who seems to know everything is Marco. And Marco is running out of hours.

The diagnostic story here is classic: growth has revealed cracks in the foundation. On paper, each project follows a familiar path—discovery, design, procurement, build, handover. In practice, every project is a snowflake because no one agrees on what “done” looks like at each stage. The sales team sells custom work without consulting the design calendar. Designs stall because procurement doesn't know the specifications it needs to order from. Subcontractors show up to job sites without the right materials because no one sent a pre-work checklist. Marco's phone vibrates constantly because he's the only human router. If you ask him what slows the business down, he'll say “everything,” but he can't tell you which bottleneck is costing the most money. That's the problem this chapter solves: a simple diagnostic to map capacity, identify constraints, and make the founder-dependency visible so you know where to start.

Think of your business as a river. Water enters as leads, flows through sales, design, delivery, and finally out as revenue and referrals. If a dam exists, the river backs up. You can add more water (more leads) or deepen the channel (more capacity), but if the dam is there, the only thing that changes is how much water spills over the sides. In a service business, the dam is often a decision only the founder makes, a step only the founder can do, or a handoff that's never written down. In product and e-commerce businesses, the dam might be inventory, supplier reliability, or a fulfillment step that can't handle peak demand. In agencies and SaaS, it's frequently a messy sales-to-delivery handoff or a custom onboarding that only a senior engineer knows how to run. Your job is to find the dam and either remove it or widen the channel around it.

To do that, we're going to run three simple tools this week. First, a Bottleneck

Checklist that surfaces where work gets stuck. Second, a Responsibility Heatmap that shows how much of the company's key work lives in the founder's head. Third, a Quick Scoring Tool that turns those observations into a single priority number. You don't need fancy software or a consultant for this; you need an hour of quiet, a whiteboard, and a conversation with your team. The goal isn't perfection. The goal is clarity about where to act first.

Here's a short story that shows how this works. Lina runs a five-person digital agency. Before running the diagnostic, she felt like everything was slow. Projects always took longer than quoted. Clients complained about long turnaround times for first drafts. Designers waited for feedback. Account managers kept clients "warm" with vague promises. When Lina used the checklist, she discovered the real bottleneck: creative briefs were inconsistent, and Lina personally rewrote every brief before work began. The brief itself was the dam. When she replaced it with a templated intake process and trained the account manager to fill it out, the average project timeline dropped from 21 days to 13, and Lina reclaimed six hours a week. Same team, same demand, different bottleneck.

Let's run the diagnostic on your business. Start by block-building thirty minutes on your calendar. Grab the person closest to day-to-day delivery and the person who manages leads or sales. If you're a solo founder, bring in a friendly peer or advisor who asks good questions. Write the words "Capacity," "Bottlenecks," and "Founder Dependencies" at the top of a whiteboard or document. We'll touch all three.

Capacity is a measure of the system's maximum output given its current resources and constraints. To get a quick read, ask these five questions and write the answers next to each other. First, what's our average weekly output of whatever matters most—projects delivered, hours billed, products shipped, onboarded customers? Second, what's our maximum weekly output if we pushed hard without breaking things? Third, which role or function consistently works overtime or is the last to finish? Fourth, where do new requests pile up—sales, design, production, support? Fifth, what's our average cycle time from "start" to "done" and has it increased over the past six months? If your maximum output isn't much higher than your average, you're running near capacity. That's not a crisis, but it means the next dam will be painful.

Bottlenecks are the specific steps or decisions that limit throughput. To find them, we'll use the Checklist of Dams. Run through the following prompts and mark "Yes" if the statement applies to your business. Yes means there's likely a bottleneck at or near that step. Don't overthink the wording; trust your gut.

- We frequently wait on a single person (often the founder) for approvals or decisions. (Yes/No)
- Work piles up at the same stage regardless of how many people are on the

- team. (Yes/No)
- Clients ask for status updates because they don't know what's happening next. (Yes/No)
  - We promise dates we miss regularly, and the reason isn't weather or illness. (Yes/No)
  - A "standard" project still requires custom work each time. (Yes/No)
  - Sales closes deals without consulting delivery, and that causes friction later. (Yes/No)
  - The founder personally rewrites or redoes significant portions of work. (Yes/No)
  - Team members ask the same questions repeatedly. (Yes/No)
  - We have no documented steps for key tasks; they live in someone's head. (Yes/No)
  - Our tools or inboxes are the only places where the next step is visible. (Yes/No)
  - Onboarding a new customer takes longer than it did a year ago. (Yes/No)
  - The same mistakes happen regularly (missed specs, wrong items, late invoices). (Yes/No)
  - We lack clear ownership; multiple people can start the same task. (Yes/No)
  - Handoffs are vague or happen verbally. (Yes/No)
  - We often "discover" midway that we're missing information or materials. (Yes/No)
  - The business depends on one person knowing where files or passwords are. (Yes/No)

Count your Yes answers. If you have more than five, you likely have multiple dams, and the founder is probably one of them. If you have more than ten, your business is a "founder-dependent hero shop," which means growth is making you less happy, not more. Don't panic. You now have a list of clues. These Yes answers map directly to places where a simple checklist, a one-page SOP, or a clear handoff would relieve pressure. For instance, if "We wait on a single person for approvals" is a Yes, that's a decision rights problem. If "Work piles up at the same stage" is a Yes, that's a capacity or skills problem at that step. If "Sales closes deals without consulting delivery" is a Yes, that's a handoff problem.

Next, create the Responsibility Heatmap. This tool shows how much of the essential work depends on you. Draw a grid. Down the left column, list the twelve to fifteen core activities that keep your business moving, for example: lead intake, qualification, proposal, contract, scheduling, design, procurement, production/fulfillment, QA, invoicing, onboarding, client updates, renewals, and hiring. Across the top, list the roles in your company—Owner/Founder, Sales, Design, Production, Ops, etc. Inside the grid, for each activity, put a checkmark under the person who currently does it most of the time or holds the responsibility. Then mark with an F any step where the founder's involvement is mandatory, even if it's not daily. Be honest. If proposals are technically assigned to sales but you rewrite every one, the checkmark goes under you.

What you're looking for is concentration. If your name has more checks than anyone else and several Fs, you are the bottleneck. If multiple boxes in a single column are empty, that role is unclear or overloaded. If one role has more checks than seems

reasonable, you may have found a hero employee you're at risk of burning out. Lina's heat map showed that the "Creative Brief" activity had a check under "Account Manager," but an F next to her name because she rewrote them. That small F told her exactly where to act. Marco's map showed the "Approval" activity had an F next to his name for every single project. That one fact explains why his phone never stops.

Now, let's create a simple score to prioritize which dam to tackle first. This is not a scientific formula; it's a quick tiebreaker. For each potential bottleneck you identified in the checklist (the steps where Yes appeared), assign three scores from one to five, with five being the worst:

- Frequency: How often does it slow us down? (1 = rarely, 5 = daily)
- Impact: How big is the delay or cost? (1 = small, 5 = expensive or dangerous)
- Spread: How many people or projects are affected? (1 = one person or project, 5 = everyone or everything)

Add the three scores together. The highest total is your best first target. Example: "Approvals" for Marco scored Frequency 5, Impact 5, Spread 5 for a total of 15. "Client updates" scored Frequency 4, Impact 2, Spread 5 for a total of 11. That tells Marco to start by fixing approvals. For Lina, "Brief rewrites" scored 5, 4, 5 (14) while "Client status requests" scored 3, 2, 5 (10). She started with briefs. This score keeps you from chasing the squeaky wheel and focuses you on the constraint that moves the business if you fix it.

Let's put some numbers behind this with a quick quantitative case. A five-person boutique strategy consultancy in Austin used these three tools in a half-day workshop. Before the diagnostic, they delivered an average of three strategy projects per month with an average margin of 32%. The founder reviewed every final deliverable and often rewrote slides; the approval bottleneck was at his desk. Their score for "Founder approval" was 5/5/5 = 15. They replaced the ad hoc review with a two-gate process: internal peer review using a rubric and a final founder sign-off only for top-tier clients or strategic accounts. Within two months, they delivered four projects per month, average margin rose to 40%, and the founder regained five hours per week. The bottleneck wasn't capacity; it was a single step with a single owner.

Another example from a different industry: a regional e-commerce business selling specialty foods. They had frequent stockouts during promotions. The diagnostic revealed the bottleneck at reorders; the founder used a gut-based purchasing model and reviewed orders weekly. Frequency 5, Impact 4, Spread 5, score 14. They implemented a simple reorder point system tied to sales velocity and a weekly procurement checklist owned by the operations manager. Stockout incidents fell from five per quarter to one, and cash tied up in excess inventory dropped by 18% in three months. The same team, same website, better flow.

A third example from a SaaS startup: a ten-person team with slow onboarding for new customers. The bottleneck was a handoff from sales to customer success, where the founder had to manually review every new account. Score 14. They created a handoff checklist and a standard onboarding path inside their product. Average time-to-first-value dropped from 15 days to 8, and churn decreased from 6% to 4% month-over-month. Again, the fix wasn't hiring or technology; it was clarifying a single step and removing a dependency.

If these numbers resonate, you might want a simple way to track the impact of your fixes. That's what the Quick Scoring Tool is for. At the end of each week, ask yourself two questions and record a simple score. First, how many decisions were blocked by waiting for the founder? Second, how many hours did the founder spend on tasks a junior could do if the process were documented? Score both from 0 to 5, where 0 is "none" and 5 is "constant." Chart these two numbers each week. As you remove bottlenecks, you should see both scores decline. A founder who reduces their decision-blocking score from 5 to 2 has moved from being the dam to being the gardener.

A helpful sidebar at this point is the Responsibility Heatmap in table form. Imagine a grid with activities listed vertically and roles horizontally, where the founder's column has checks at "Proposal," "Approval," "Scheduling," and "Hiring," while the Ops Manager column is empty. That visual alone is enough to spark a change. You don't need to formalize this into a complex RACI yet; you just need to see where the weight sits. You'll formalize decision rights later in Chapter 10. For now, awareness is enough.

There are three common missteps to avoid when running this diagnostic. First, blaming people instead of processes. If multiple people make the same mistake, it's a process or training problem. Second, mistaking a tool for a fix. Buying a new CRM doesn't fix an unclear handoff; it just moves the mess to a new platform. Third, solving a symptom. If clients complain about slow responses, the symptom is slowness; the real problem might be that nobody knows who owns "client updates." Fix the ownership and the speed follows. Keep your eye on the constraint.

To make this immediate, here are two short exercises you can do this week. The first takes thirty minutes. List your top five recurring projects or products. For each, write down the last three steps that always seem to slow you down. For each step, ask: Who must be involved? What information is missing? What decision is required? For any step where your name appears as the required decision-maker, mark it with a star. That step is a bottleneck candidate. At the end, pick the step with the most stars and score it using the Frequency-Impact-Spread tool. This becomes your first target.

The second exercise takes an hour and should be done with your team. Go around the room and ask each person: What task do you spend the most time waiting on? Then ask: What one checklist or template would save you an hour a week? Write the answers on a whiteboard. You'll see patterns. If three people say "waiting for approval

on change orders,” you’ve found your next target. If two people say “client asks for status,” you have a visibility problem. For each pattern, apply the quick score. At the end, agree on the single highest-scoring item and write a one-sentence problem statement: “We lose two days per project waiting for founder approval of scope changes.” That’s your starting line.

Here’s a mini-template to capture your diagnostic in one place. Think of this as a one-page brief you can keep visible.

- Business name and date:
- Average weekly output (projects/invoices/shipments):
- Maximum weekly output (pushing without breaking):
- Where work piles up (stage or role):
- Bottleneck Checklist Yes count:
- Top three Yes items:
- Responsibility Heatmap observation (where founder checks dominate):
- Quick Score of top bottleneck (Frequency + Impact + Spread):
- Proposed first target fix (one sentence):
- Owner of the fix (not the founder, ideally):
- Success metric for the fix (what will change in four weeks):

You can copy this into a document and fill it out in under twenty minutes. The value isn’t in perfection; it’s in having a single page that tells you where to focus.

If your business has both service and product elements, or if you run a marketplace or hybrid model, you may need to run the diagnostic on each stream separately. A common pattern in hybrid businesses is that one stream hides the bottlenecks of the other. For example, a firm that does both consulting and productized training may find the consulting side absorbs all the founder’s time, while the training side suffers from neglect and underperforms. Split the diagnostic: list the activities for the service side, then for the product side, and map responsibilities separately. You might discover the training side is actually more scalable and deserves more resources.

For the visually inclined, here’s a simple flow you can sketch to guide your diagnostic work. Imagine a horizontal flow from left to right: Leads → Sales → Intake → Delivery → Billing → Outcomes. Under each stage, write the current bottleneck you suspect, and next to it, write the person who owns it. If the word “Founder” appears in more than one box, that’s a clear signal. If the same person appears in multiple boxes in a way that doesn’t make sense (e.g., Sales also does Delivery), that’s a second signal. Your goal is to reduce the number of boxes that say “Founder” and to align boxes so that one person doesn’t own consecutive steps unless they truly should.

A common question at this point is, “What if the bottleneck is that we don’t have enough leads?” That’s a capacity problem in sales, and it might be the right place to focus. But first, confirm whether you’d be able to handle a surge if leads doubled tomorrow. If your Delivery or Billing would collapse, then the real constraint is inside

the business, not out front. Fix the internal dam first, then pour water in. This sequence prevents the classic pattern of selling more than you can deliver, which damages reputation and burns out the team.

If you're a solo founder, this diagnostic still works. Replace "team" with "you and your contractors or part-time help." Your heatmap will be stark, and your bottleneck checklist will mostly have your name on it. That's fine; this book is written with you in mind. The path out starts with documenting what you do, then delegating the repeatable parts to trained help. In practice, solo founders often find that 60-70% of their week consists of tasks that could be done by a trained assistant or a junior hire if the steps were written down. That's your first target.

Before we wrap this diagnostic, let's acknowledge a simple truth: the numbers you get today are a snapshot, not a sentence. They reflect your current capacity, processes, and roles. As you change those, the numbers will change. The Quick Scoring Tool is meant to track progress, not to make you feel bad about the current state. If your founder-blocking score is 5 this week, great. If it's 4 next week, that's progress. Celebrate it. You're turning a dam into a gate you control, and that's the core of scaling.

Finally, a quick note on courage. Many founders resist this diagnostic because it makes the dependency obvious. If you see yourself as the bottleneck, it can feel threatening. But it's also liberating. Acknowledging the dam is the first step to removing it. The businesses that scale are the ones that face the constraint and fix it, not the ones that add more paddlers to a canoe with a hole. You're about to build a system that lets you lead instead of lugging.

Here's the implementation plan for this week, step by step. First, block a 90-minute session tomorrow to run the Bottleneck Checklist by yourself. Mark Yes where it applies and write a one-sentence description of the situation next to each Yes. Second, schedule a 60-minute meeting within two days with your direct reports or a trusted peer. Bring the Yes list and use the last twenty minutes to co-create the Responsibility Heatmap. Third, before the meeting ends, score your top three candidates with the Frequency-Impact-Spread tool and agree on the single highest-scoring target. Fourth, draft the one-page diagnostic template and share it with the team so everyone sees the same picture. Fifth, create a simple tracking note where you'll write the two weekly scores (decision-blocking and junior-task hours) for the next eight weeks. That's it. Five steps, all doable this week, all producing clarity you can act on.

As you complete these steps, keep your language neutral and specific. Instead of "everything is slow," write "approvals take 2-3 days." Instead of "sales is a mess," write "proposals wait 48 hours for founder rewrite." Specificity turns complaints into fixable problems. The goal is to leave the week with a single, prioritized target, a clear statement of its impact, and an owner who isn't you. That's the foundation for

everything that follows in this book.

One last check: if you complete the diagnostic and nothing feels urgent, consider whether you're currently coasting on overwork. If you're the founder and you're working 60+ hours but the business feels fine, your capacity might be artificially high because you're the universal shock absorber. In that case, your bottleneck is yourself at a lower frequency but higher risk. The diagnostic will still surface it. The fix is the same: document, delegate, and remove your mandatory involvement from non-critical steps.

You now have a map. You know where the water is backing up. You have a single target to fix and a way to measure whether the fix is working. In the next chapter, you'll take that target and design a process that's outcome-first, not task-first, so the fix sticks and the system gets stronger as you add people. For now, close this page, open a blank document, copy the one-page diagnostic template, and start with the Bottleneck Checklist. The sooner you see the dams clearly, the sooner you can start taking them down.

SAMPLE COPY

---

*This is a sample preview. Purchase the book to read the full content.*

Visit [MixCache.com](https://MixCache.com) to purchase the complete book.

SAMPLE COPY