

The PACE Process

Practice · Adjust · Calibrate · Extend

How leaders and teams get measurably better at the work they already do

A Practical Toolkit for Building Adaptive Capability at Scale

Version 6.0 · For CEO and Senior Leader Use

Bill Flynn · Catalyst Growth Advisors

Toyota's Stolen Plans

Toyota's competitors have often gone to remarkable lengths to get inside the system. Mike Rother describes some of this in *Toyota Kata*. They toured the plants, hired away the managers, and reverse-engineered the methods. The story goes that even when competitors got their hands on Toyota's plans, Toyota was not particularly worried. By the time anyone implemented what was on those pages, Toyota had iterated beyond them. The advantage was never what was written down. It was the way Toyota had wired learning into how everyone worked. Every day.

PACE is how you build that for your own organization. Run it for a year and you will:

- Run the PACE loop weekly with your direct reports, turning ordinary decisions into compounding judgment.
- Diagnose why something did not work well enough to fix the right thing next time, rather than repeating the same mistake with more effort.
- Build a small, shared set of notes on what works in your organization, written by the people doing the work.
- Spot patterns across your leadership team before they become performance problems.

If the promise of this toolkit could be reduced to one sentence: your organization learns from its own experience faster than the environment is changing it.

IMPORTANT: PACE has two working parts. The PACE loop is what you run every week. The PACE card is where you capture it. Everything else in this document supports those two things.

Why This Works: The AlphaGo Zero Insight

In 2017, DeepMind built an AI that learned to play Go better than any human or any previous AI. What was unusual was how it learned. The earlier version, AlphaGo, had been trained on millions of expert

human games. AlphaGo Zero was given only the rules and left to play itself. It rapidly surpassed the expert-trained version in every measure.

The breakthrough was not better rules. It was a better learning architecture.

Most leadership development is built like the original AlphaGo: encode what experts know, train people on the encoding, hope they execute. PACE is built like AlphaGo Zero. Instead of encoding expertise into procedures, it gives leaders a structured way to generate expertise from their own experience and refine it over time. The goal is not to scale knowledge. It is to scale the capacity to learn.

This is a reference to learning architecture, not AI adoption. PACE is not a toolkit for deploying AI inside your company. It is a toolkit for improving the judgment of the humans running it.

Where the analogy breaks is as important as where it holds. Leaders do not have perfect information, instant reward signals, or millions of iterations per day. Politics, ego, turnover, and slow feedback distort the signal. That is not a reason to abandon the principle. It is the reason the loop in this toolkit is slower, more deliberate, and more dependent on external signal and calibration than a self-play algorithm ever needs to be. Read the comparison as a principle, not a protocol. The principle stands. The implementation has to absorb the friction the principle ignores.

What PACE is, plainly

PACE is the practice of continual, disciplined, accelerated discovery. Continual because the loop runs every week, not when something breaks. Disciplined because the cycle runs the same way each time, with a written rule, a target condition, and external signal. Accelerated because each cycle compounds on the last, and a year of cycles produces judgment that no single year of experience could.

The stake, in arithmetic

Picture a typical scaling company. One CEO or senior executive spending roughly 25 hours a week solving problems that someone closer to the work could solve. Five direct reports doing the same. Twenty-five team leads underneath them doing the same. That is 775 hours of the organization's week spent on work that does not require the level of the person doing it.

Assume 80 percent of that time is recurring problems, the same issues surfacing in new disguises because nothing in the system changed the last time. That is 620 hours a week. At a 40-hour week, 15 full-time salaries' worth of capacity being consumed by work that should not be reaching those people in the first place.

The numbers are yours to adjust. Change the layer count, the hours, the percentage. The arithmetic still lands in the same place. This is not a rounding error.

There is a second lens worth naming. Decisions that get made at the right level are cheaper as well as faster. A senior team making four weeks of decisions in two weeks compounds differently than one

making them in four. The work downstream of a decision cannot start until the decision is made. Velocity upstream is velocity everywhere.

PACE is the architecture that moves decisions to where they should be made. The AlphaGo Zero principle is what makes it work. The arithmetic is what makes it matter.

How PACE Scales

PACE is one method that runs at every scale. An individual leader runs the PACE loop on rules about their own behavior. A team runs the PACE loop on rules about how the team works together: each member practices, each captures on a PACE card, the team compares notes weekly. A division runs the PACE loop on rules about how work flows across teams. An organization runs it at the widest scope, with the CEO sponsoring and governance keeping the shared rule set current.

What changes across scales: the scope of the rule, the cadence of the cycle, and the coordination required to run it. What stays constant: Practice, Adjust, Calibrate, Extend. The structure is the same at every level.

Who This Is For

CEOs, senior executives, and team leaders of scaling organizations where judgment matters more than compliance, situations vary too much for a fixed playbook, and the job has shifted from having the answers to building the systems that generate answers.

This is written for companies that want to scale learning. The language, the cadences, and the governance assume an organization small enough that the CEO can still influence what gets practiced and large enough that they cannot make every decision themselves. The method works at any size above that threshold. At larger companies the rollout is staged across divisions and the CEO sponsors rather than runs the cycles, but the mechanism is the same.

When PACE is the wrong tool

Before adopting, confirm none of the following describes your situation:

- Routine, low-variability work where precision and consistency are the primary goals. A warehouse, an assembly line, a claims-processing team. Standard work and training is the right intervention there.
- Crisis response with no slack. PACE requires space to reflect between iterations. A team fighting a live crisis should not be asked to also run the PACE loop on how they are handling it. Debrief afterward, not during.
- Teams with critically low psychological safety. The external signal step depends on honest feedback. In teams where that is unavailable, PACE will produce confidently wrong rules and

add them to the shared rule set. Safety work comes first. See the Getting Started Guide for the safety check-in.

- Organizations that already have too many operating cadences competing for attention. Adding PACE on top of an overloaded system will not work. Replace or retire something before adopting.

Time investment

Cadence	How much time
Weekly PACE loop (per leader)	30 to 45 minutes total. About 20 minutes of structured reflection on a single page, plus a 10 to 15 minute conversation with one other person to check what they observed. The reflection happens inside work the leader is already doing.
Weekly coaching (per direct report)	15 to 20 minutes per direct report, once a week. A short structured conversation about what they are working on and what they are learning.
Monthly team rule review	A 30 minute meeting, plus 15 to 20 minutes of prep for the two people presenting.
Quarterly model review	60 to 90 minutes. The leadership team reviews what is working, what to retire, and what to test next.
First-pass manager enablement	A half-day orientation for direct managers running cycles with their teams. One-time, before rollout begins.

For a leadership team of six or eight running PACE together, the weekly investment is roughly four to six hours of distributed time across the team. That is a small fraction of the hours those leaders already spend leading, but it will not fit on top of an already-overloaded calendar without something giving way. Part of adopting the method is deciding what you will retire to make room for it. A recurring status meeting is a common candidate.

How to Use This Document

If you have not yet run a cycle yourself, start with the Getting Started Guide on the next page. One cycle, one week, in your own work. Run it for two to four weeks before you ask anyone else on your team to do the same. If you are ready to install PACE in your leadership team, read Parts One through Four and then turn to Installing PACE in Your Leadership Team. If your work is genuinely complex, where feedback

is slow or misleading and the right answer is only visible in retrospect, skip ahead to Using PACE When the Environment Is Wicked for the modified method.

Getting Started Guide

You can read this document end to end before you run a cycle, or you can start this week. This guide is for the second option. It walks you through running one complete cycle with every load-bearing element in place. The rest of the document is the deeper reference behind what this guide walks you through.

Set aside about 90 minutes for the setup. The cycle itself runs across one week.

Before you start

Two gates. Pass both before moving to setup. Fail either and the method will not produce the signal you need.

Gate one. Is your work wicked?

Most leadership work is complicated, not wicked. Complicated means hard but knowable: you can analyze your way to a reasonable answer, and when you try something, you find out fairly quickly whether it worked. Wicked means the answer is only visible in retrospect, feedback is slow or misleading, and the problem keeps shifting under you as you engage with it.

A quick test. Think about a decision your team will make in the next month. If you can predict, with reasonable confidence, whether a given approach will work, the environment is complicated. If your best honest answer is “we will find out,” the environment is closer to wicked.

If your work is wicked, this guide will not apply cleanly. Turn to Using PACE When the Environment Is Wicked for the modified method. If your work is complicated, continue.

Gate two. Is the team safe enough?

The loop depends on people telling you the truth about their experience. If they will not, you run cycles that produce false signal and promote it into your judgment.

If you are running the cycle with your team, run the four-item safety check-in below. Anonymous. 1 to 5 scale. Average the scores.

On this team I can, without fear:

- Speak up freely and admit mistakes.
- Take interpersonal risks.
- Raise problems and tough issues.
- Ask others for help.

If the team averages 4.0 or above, proceed. If the team averages 3.0 to 3.9, proceed cautiously. Your first rule should be safety-adjacent: speak last in group settings, ask before telling, name your own uncertainty out loud. The loop itself becomes a safety-building intervention. If the team averages below 3.0, stop. Address safety first. The method will not work and may make things worse. Re-measure in 60 to 90 days.

If you are running your first cycle on yourself before asking anyone else to run one, the gate is simpler. The one person you will ask for external signal has to be willing to tell you the truth. If you can name that person right now, proceed. If you cannot, pick a different person or fix the relationship first.

Setting up your first cycle

Six setup decisions. Each one is a small act of writing something down. Together they determine whether the cycle produces signal or noise. Most first-cycle failures happen here, not in the run itself.

What is a rule?

A rule is a single observable behavior a person does or does not do in a specific situation. It is drawn from someone who consistently produces the outcome you want, and it is specific enough that an outside observer could tell you whether it happened. The rule you bring to the cycle is a reasoned hypothesis, not a guess. You have seen this behavior work, in yourself or in someone whose results you trust, and you have reason to believe practicing it will move the system. The cycle is what tests whether you are right.

A rule looks like this: “Before stating my own view in a leadership team discussion, I ask each direct report what they see.” Not this: “Be more open to input.” The first is a behavior. The second is a disposition. PACE works on behaviors because behaviors are testable and dispositions are not.

1. Pick the domain

One area where progress would be observable in 30 to 60 days. Choose where the pain is most visible right now. Decisions that get re-litigated. A team that escalates too much. Priorities that do not stick. Meetings that produce no decision.

Your domain: _____

2. Identify a source for your rule

A rule picked from thin air is a guess. A rule picked from a valid source is a hypothesis with grounding. You need one or both of these:

Internal exemplar. Name two or three people in your organization who consistently produce good outcomes in this domain. Not the loudest person. Not the person with tenure. Someone whose results hold across multiple situations. Watch for selection bias. If your three exemplars all look, talk, and think alike, widen the list before extracting a rule.

External source. Published research in the domain, a book you trust, or a practitioner you respect who has solved this problem before. A well-constructed AI research prompt can surface external sources quickly. Treat AI output as hypothesis material, not truth.

***One caution on internal exemplars.** Reputation is not performance. Before you build a rule from someone's behavior, watch them work in the domain you picked. Sit in one of their meetings. Observe one of their one-on-ones. What people do in the room often differs from what they say they do.*

Your source or sources: _____

3. Pick a candidate rule

One observable behavior from the source. If you have a rule you already trust from experience, use it. If you need a starting set, Starter Rules You Can Practice has three: the 5Cs for decision quality, the 6As for team performance, and execution heuristics for operational drag. Pick one that matches your domain.

Your candidate rule: _____

4. Run the rule through the gauntlet

Five quick checks. Each is a yes or no. If any fails, refine the rule or pick a different one before moving on.

- **Behavioral.** Can you watch someone do it or fail to do it?
- **Testable.** Could you tell after 30 days whether it was working?
- **Portable.** Would it hold up for a different person in a different context?
- **Survives the outlier test.** Would high performers outside your organization converge on something similar?
- **Rule not value.** "Be customer-focused" is a value. "Start every customer conversation by confirming the problem they are trying to solve" is a rule.

If your rule passed all five, keep it. If it failed any, revise and run it through again. The ten minutes you spend here saves the four weeks a bad rule will waste.

Your final rule: _____

A note on order

You will write your rule before you write your target condition. This is intentional. The problem you are working on is already known. The rule is a reasoned hypothesis about what behavior, if practiced, would move the system, built from your experience and whatever else you can bring to bear. You are not committing to the rule being right. You are committing to running the cycle honestly enough that you

will see quickly whether it is working, and you will replace it with a better one if it is not. The target condition is what makes that visible. It is the downstream outcome the rule has to earn.

The risk in this order is writing a target condition that simply describes what your rule will produce, which guarantees the rule looks successful and tests nothing. If your rule is “ask each direct report what they see before stating my own view,” your target condition is not “I will hear from each direct report.” That is just the rule. Your target condition is something like “my direct reports generate the majority of new ideas in our weekly meeting within eight weeks.” That is testable and it can fail.

5. Name the target condition

The rule is the behavior you will practice. The target condition is the observable state you are trying to reach by running the cycle. Two rules for target conditions. They must be observable by someone other than you. And they must have a date. Without both, you have an intention, not a condition.

Examples.

Rule: Separate facts from assumptions before generating options in decision meetings.

Target condition: By the end of month two, at least eight of ten decisions in our weekly leadership meeting end without being re-litigated in the following two weeks.

Rule: Speak last in group settings.

Target condition: Within six weeks, direct reports propose solutions before I do in at least three of four team discussions.

If you cannot name the target condition, you do not yet know what the loop is for. Stop and work on this before continuing. This is the most common early-stage failure mode.

Your target condition: _____

6. Pick the situation and the person

One specific upcoming event where you will practice the rule. Not “this week” or “in meetings.” A named day, meeting, and set of people. The Tuesday budget review. The Thursday one-on-one with Sarah. The Monday priority meeting.

One colleague who will be in the situation and will tell you the truth about what they experienced. Someone who will answer honestly when you ask about their experience, not someone who will tell you what they think you want to hear.

The situation: _____

The person: _____

Before you run: the six-item readiness check

All six should be filled in before you run the first cycle. If any of the six is blank or vague, return to that step and sharpen it. Running a cycle on an incomplete setup is the most common way PACE produces false signal.

- Domain is named.
- Source is named (internal exemplar or external source or both).
- Candidate rule is written in one sentence.
- Rule passed all five gauntlet checks.
- Target condition is written with an observable outcome and a date.
- Situation is a specific day, meeting, and set of people. Person is named.

If all six are checked, you are ready to run the cycle. If even one is blank, fix it before proceeding. This is the difference between running PACE and running something thinner that wears the name.

Running the cycle and closing it

You have completed setup. Now run the cycle. One week.

Run the rule in the situation

Apply the behavior in the actual meeting, decision, or conversation. Not in a simulation. Not in a rehearsal. Real work, real stakes.

Within 24 hours, capture everything on the PACE card

Fill in the Before section before the event. Fill in the After, Signal, and Next sections immediately after. Keep it to one page. The discipline of writing it down is part of what makes the loop work.

THE LEARNING LOOP CARD	
BEFORE: Set your intent	AFTER: Capture what happened
Rule (hypothesis you are practicing): Target condition (observable state, with date): Control (leading indicator, behavior you can see weekly): Consequence (lagging indicator, business outcome):	Where I followed the rule: Where I drifted: What happened: Progress toward target condition: My failure diagnosis:

THE LEARNING LOOP CARD	
Situation: Specific commitment:	
SIGNAL: What others experienced	NEXT: One adjustment
Person asked: What they said: Pattern (same as last time?):	One specific change: Next situation to practice in:

A worked example of a filled card is in The PACE Loop in Depth. A standalone printable version of this card is available as a separate file.

Work through the After section in order. Where did I follow the rule? Where did I drift? What happened as a result? If the outcome was poor, which failure type: wrong rule (the rule itself does not apply here), wrong execution (the rule was sound, you applied it poorly), or wrong context (the rule is sound and was well executed, but this situation was outside its domain)? Each failure type calls for a different adjustment.

Ask the one person the experience question

Not “how did I do.” “What was your experience of that conversation?” The first question gets evaluative answers. The second gets data. Write down what they say exactly. One person can be discounted. Two or three saying the same thing becomes signal.

Name one adjustment for the next cycle

One specific change, stated as a behavior. “Next time, I will ask the dissenting question before we start building consensus.” Not “I will be more thoughtful.” The next cycle practices the adjustment.

Repeat within seven days

Same rule, or a refined version based on what you learned. The cycle compounds. Speed and consistency are the advantage.

After three or four cycles, a check-in

Stop and answer four questions in writing. Ten minutes.

- What rule am I working on?
- What have I learned so far?
- What is my diagnosis: wrong rule, wrong execution, or wrong context?
- What is my one adjustment for the next cycle?

If you cannot answer the second question with anything specific, the loop has not been running long enough or you have not been ruthless enough about external signal. Most leaders default to evaluating their own performance and call it learning. The check-in surfaces the gap.

Deciding whether to persist, refine, or replace

The goal is calibrated iteration. Not stubborn persistence and not premature abandonment.

Look for a pattern, not a count. A pattern is three consistent results across varied contexts and different people. One person, one situation, one bad outcome is not a pattern.

***One critical qualifier.** Attempts where you drifted significantly from the rule do not count toward the pattern. A poor result from poor execution is not evidence about the rule. Only count attempts where your execution was solid.*

After three to five solid attempts in a complicated environment, you should see a pattern emerging. If results are mixed or improving, keep iterating. If a clear negative pattern is emerging despite solid execution, examine the rule. If the pattern holds across varied contexts and multiple people confirm the same experience, replace the rule.

The PACE Loop in Depth has the full persist-versus-abandon logic, including the pattern thresholds for wicked environments and the specific signals that distinguish each decision. Come back to it when you need more detail.

What comes next

You are now running PACE. One cycle a week in one domain. Keep going for four to six weeks before adding anything else.

If you want to deepen your understanding of what you just did, read *The PACE Loop in Depth* next. It is the reference for the cycle, the card, the failure diagnosis, and the patterns.

If you are ready to install the method in your leadership team, read through the rest of the document and then turn to *Installing PACE in Your Leadership Team*.

Running one honest cycle will teach you more than reading will.

PART ONE

The PACE Loop in Depth

This is the engine of the method. The PACE loop is how judgment improves. Not through training, not through reading, but through repeated real-world application with structured reflection. The speed and consistency of the loop is the advantage.

Standardize the loop. Customize the rules. The mechanism for improvement is universal. What you are improving is specific to your context.

PACE labels what is happening at each step of the loop: Practice covers Steps 1 and 2, Adjust covers Steps 3 through 5, Calibrate is Part Three, Extend is Part Four. The naming scales. A CEO can ask whether the company is running PACE at the board level. A team leader can answer at the loop level.

Name the Target Condition Before You Run the Loop

The Guide introduces the target condition and gives two worked examples. The deeper question is what makes a target condition hold up under real work.

Good target conditions share three qualities. They are observable by someone other than you, which means behavior or outcome changes that could be confirmed independently, not a private sense that things feel better. They have a date, which is what turns an intention into a condition. And they pair a control with a consequence: a leading indicator you can see weekly, and a lagging indicator that tells you whether the behavior is producing the business outcome you care about.

The most common failure is writing a target condition that measures the behavior twice. “By month two, I will consistently ask the dissenting question before consensus builds” is not a target condition. It is a restatement of the rule. The target condition has to specify what changes in the system once the rule is running.

A second common failure is a target condition whose lagging indicator lags too far. If the only measurable consequence shows up in twelve months, the loop has no interim signal. Choose a consequence that can be read within a quarter, even if the fuller effect takes longer.

Prerequisite: The Psychological Safety Pre-Check

The four-item safety check-in and the three-band scoring guidance are in the Getting Started Guide. Before running cycles with a team, do not skip that check. The external signal step depends on people telling you the truth about their experience. A team that will not will produce confidently wrong rules, and those rules get promoted into the shared set. Below-3.0 averages mean stop. 3.0 to 3.9 means proceed cautiously with a safety-adjacent first rule. 4.0 and above means proceed. Re-measure every 60 to 90 days; safety erodes faster than it builds.

The Single Biggest Failure Mode: The Leader Reintroducing Dependency

If PACE fails inside a team where the basic preconditions are met, the most likely reason is this: the leader who adopted the method also undermines it by continuing to solve problems personally instead of letting the loop work. Founders and high-agency leaders are particularly prone to this because problem-solving is the reflex that made them successful.

The tells are specific. If you are the leader, watch for these in yourself:

- You answer a direct report's question before they finish formulating it.
- You review their rule and rewrite it "more clearly," replacing their language with yours.
- You correct their failure diagnosis, wrong rule, wrong execution, wrong context, before they have worked through it themselves.
- You present your own loop as the template and unconsciously expect theirs to match.
- You skip the external signal step in your own loop because you already know what people will say.
- You field questions from their team that should be going to them.
- You reflexively step in when you see them about to do something you would do differently.

The antidotes are behavioral, not attitudinal.

- Ask three questions minimum before jumping in. For example: "What have you already tried?" "What did you learn from it?" "What do you think the next move is?"
- Let their rule stand in their words, even if you would phrase it differently. The rule they own is more valuable than the rule you would write.
- Run your own loop visibly and imperfectly. Share the loops where your rule did not work. The modeling is the teaching.
- Redirect questions from their team back to them, in their presence. "That is a great question for your manager. Let us bring her into this."

If you cannot change these behaviors in yourself, the method will not produce the distributed judgment it promises. Your team will learn that running the loop is optional when you are in the room. That is the signal that kills PACE faster than any other.

The Six-Step Loop

Steps 1 and 2 are Practice. Steps 3 through 5 are Adjust. Step 6 is Repeat, the discipline that makes both compound.

Step	What to do
Step 1: Intent	Name the rule you are working on. Name the target condition you are working toward, with a date, and the specific situation. The Tuesday budget review or the Thursday one-on-one with Sarah. The behavior is equally specific: in the Tuesday budget review, separate facts from assumptions before discussing options. The target is observable and timed: by month end, assumptions are named explicitly in at least three of four weekly meetings.
Step 2: Action	Apply the rule in real work, in an actual meeting, decision, or conversation, not in a simulation or workshop.
Step 3: Self-Assessment	After the action, work through these questions in order. Where did I follow the rule? Where did I drift? What happened as a result? If the outcome was poor: was the rule wrong, was my execution wrong, or was this situation outside the rule’s domain?
Step 4: External Signal	Ask one or two others what their experience was. Not “how did I do?” but “what was your experience of that conversation?” You are asking for data, not evaluation. One person can be discounted. Two or three saying the same thing is signal. If psychological safety is low, ask in writing, use a third party, or ask about the situation rather than yourself. Measure behavioral change and results.
Step 5: Adjustment	Make one specific behavioral change for the next iteration. For example: “Next time, I will ask the dissenting question earlier, before we start building consensus.”
Step 6: Repeat	Iterate quickly. The loop compounds. The faster and more consistently you run it, the faster judgment improves. Aim for at least one full loop per week in your chosen domain.

The loop draws on several converging research streams: Ericsson’s deliberate practice, Argyris and Schön’s double-loop learning, and the NeuroLeadership Institute’s AGES model on how the brain consolidates learning.

The PACE Card

The blank card template is in the Getting Started Guide. A filled example follows below. A digital version, a template in Notion, Airtable, or whatever system your team already uses, works equally well. Consistency of practice matters more than the medium.

A filled card: what good looks like

The card below shows one week of a loop aimed at meeting discipline, a challenge most leadership teams recognize. Your first card will not be this clean. Use this as a reference for what the fields are asking for and what a complete entry looks like.

THE LEARNING LOOP CARD	
BEFORE: Set your intent	AFTER: Capture what happened
<p>Rule (hypothesis you are practicing): End every agenda item with a named owner and a specific next action before moving to the next item.</p> <p>Target condition (observable state, with date): Within eight weeks, every agenda item closes with owner and next action in writing, and action items are completed on time or re-surfaced with an explicit reason.</p> <p>Control (leading indicator, behavior you can see weekly): Percentage of agenda items that close with a named owner and a specific next action captured in the shared meeting doc. Target: 90 percent by week eight.</p> <p>Consequence (lagging indicator, business outcome): Percentage of action items completed by the next meeting, and number of items re-surfacing without a stated reason. Target: 80 percent on-time completion; fewer than 15 percent of items re-surfacing unexplained.</p> <p>Situation: Thursday's exec staff meeting, 9 to 11 AM, eight-person leadership team, seven agenda items.</p> <p>Specific commitment: Before moving on from any agenda item, I will stop and ask two questions out loud: who owns this, and what is the specific next action by when. I will write both answers in the shared meeting doc before the next item starts.</p>	<p>Where I followed the rule: On five of the seven items, I stopped the discussion and captured the owner and next action before moving on.</p> <p>Where I drifted: On the two longest items (budget reforecast and the customer escalation), discussion ran over and I let the conversation shift to the next topic without closing. By the end of the meeting neither had an owner in writing.</p> <p>What happened: The five items that closed with owners produced follow-up work within 48 hours. The two items without owners showed up unresolved on next week's agenda. Net meeting time went up by about 15 minutes because of the pauses, but three items that would normally have bled into next week got finished.</p> <p>Progress toward target condition: Control: 5 of 7 items closed with owner and next action (71 percent versus 90 percent target). Consequence: too early to measure; earliest read will come at next week's meeting when this week's action items come due.</p> <p>My failure diagnosis: Wrong execution. The rule is right. I drifted on the items where the discussion was most interesting, which is exactly where the rule is most needed.</p>
SIGNAL: What others experienced	NEXT: One adjustment

THE LEARNING LOOP CARD	
<p>Person asked: The head of operations.</p> <p>What they said: “I noticed you were closing items this week. It felt different. I left the meeting knowing what I was going to do next, which is honestly new. On the two that did not close, I was not sure whether we had decided anything or if it was mine to push forward. I would rather you stop us on those than let them drift.”</p> <p>Pattern (same as last time?): First cycle, no pattern yet.</p>	<p>One specific change: Next Thursday, I will put a two-minute buffer at the end of every agenda item in the shared doc, explicitly reserved for naming owner and next action. The buffer is the forcing function, not my memory in the moment.</p> <p>Next situation to practice in: Next Thursday’s exec staff meeting.</p>

Diagnosing What Went Wrong

When a rule does not produce the expected outcome, the right adjustment depends on correctly identifying what failed. There are three possibilities.

A cycle that surfaces a wrong rule is not a cycle that failed. It is the loop doing what it was designed to do, which is to make a wrong hypothesis visible quickly enough to replace it with a sharper one.

Failure type	How to identify it and what to do
Wrong rule	The rule itself is flawed or does not apply in this context. The signal: you executed well but outcomes were still poor, and multiple people in multiple situations confirm the same result. Action: refine or replace the rule.
Wrong execution	The rule is sound but was applied poorly. The signal: you know what to do, but the actual behavior was inconsistent, rushed, or incomplete. Action: practice the execution, not the rule.
Wrong context	The rule is sound and was well executed, but this particular situation falls outside its domain. The signal: it worked most of the time but not here, and the situation was genuinely unusual. Action: define the boundary of the rule more precisely.

Structured diagnosis: four questions before you decide

Rushing to diagnosis is the most common failure in the failure analysis step itself. Before you commit to a diagnosis, work through these four questions in order. The discipline is borrowed from A3 thinking, a structured problem-solving practice from the Toyota Production System that documents a problem on one page by naming the gap, the hypothesis, and alternative causes before settling on action. Compressed here to fit an executive loop.

- **What was the observed gap?** Describe specifically what happened versus what the rule predicted. Keep it to one sentence. “The meeting ended without a named owner, even though the rule says to name one” is specific. “It didn’t go well” is not.
- **What is your current hypothesis about the root cause?** Name your best guess. This is your starting diagnosis, not your final one.
- **What alternative explanations did you consider?** Name at least two that are not your leading guess. If you cannot name two, you have not done the diagnosis. The most common single failure in this step is settling on the first explanation that fits.
- **Which adjustment addresses which cause?** Be explicit. “I will do X to address cause Y” is a testable next step. “I will try harder” is not.

This structure does two things. It forces the leader to articulate a hypothesis about the failure, which is itself subject to the next loop’s scrutiny. And it surfaces alternative explanations that self-assessment alone usually misses, because the reflex is to confirm the first diagnosis and move on.

When to Persist and When to Abandon a Rule

The Guide covers pattern-not-count and the solid-execution qualifier. The environment-specific adjustment, the difference between complicated and wicked domains, is what this section adds. The same pattern thresholds do not apply when feedback is slow.

Context	Practitioner heuristic for attempts
Complicated environment	Look for pattern after 3 to 5 solid attempts. Examine the rule if no clear signal after 8.
Wicked environment	Look for pattern after 6 to 8 solid attempts. Examine the rule if no clear signal after 15.
Either environment	Three consistent results across varied contexts and different people is a meaningful pattern. One person, one situation, one bad outcome is not.

Signal	What to do
Keep iterating	Results are mixed or improving. Execution has been inconsistent across attempts. You have not yet applied it in genuinely varied contexts.
Examine the rule	A clear negative pattern is emerging: three consecutive poor outcomes in complicated environments, or six in wicked ones, despite solid execution. Or roughly 75 percent or more of attempts producing the same negative signal across different situations and different people.
Replace the rule	The pattern holds even when execution is strong and context varies. Multiple external sources confirm the same experience. The rule may be locally invalid or simply wrong.

The key question is not how many times you have tried. It is whether a pattern is emerging that is consistent enough to be meaningful and varied enough not to be explained by a single bad execution or an unusual context.

The goal is calibrated iteration. Not stubborn persistence and not premature abandonment.

APPLY IT NOW. Write your answers below before moving on. These are starting points for a team conversation. 1. Write your first PACE card entry. Rule: 2. Write your target condition (what change would you expect to see, and by when): 3. Name the specific upcoming situation you will practice it in (day, meeting, or conversation): 4. Name one person you will ask for external signal afterward: 5. Write the exact question you will use to ask them:

PART TWO

Starter Rules You Can Practice

Before you can run a cycle you need a behavior to practice. Most leaders need a framework to generate one. **If you already have a decision-making framework you trust, use it. If you already have a way of diagnosing team performance, use that. The 5Cs and 6As in this section are offered for leaders who do not.** The execution heuristics are for a third domain, operational drag, where the issue is discipline more than diagnosis.

The loop in Part One is the method. The frameworks below are starting points so you do not have to build your own from scratch before you have run a single cycle.

The 5Cs Decision Framework

For decision quality. The 5Cs are Context, Clarity, Create choices, Choose, Commit. Most decision errors are not failures of intelligence. They are failures of process. The 5Cs slow down pattern-matching at the points where it most often goes wrong.

Phase	What it does
Context	What is happening right now? What has changed? What is the system we are deciding inside?
Clarity	What is the real question? Separate facts from assumptions. Name what is known versus what is believed.
Create choices	Generate real alternatives beyond yes or no. Narrow framing is the most common decision error.
Choose	Make the decision against explicit criteria, not against whichever option was proposed first.
Commit	Name the owner and the next specific action. If there is no owner and no action, there is no decision.

The 6As Team Framework

For team performance. The 6As diagnose why a team is or is not producing, before attributing performance problems to individuals.

Dimension	What it measures
Architecture	Is the team structured correctly for the work? Right roles, right interfaces, right information flow?
Alignment	Does everyone agree on goals, priorities, and what good looks like?
Agency	Do people have the authority to act without waiting for permission?

Dimension	What it measures
Accountability	Are commitments clear, tracked, and consequential?
Adaptability	Can the team change direction when conditions change, or does every pivot require the leader to drive it?
Acceleration	Is the team getting faster and better over time, or is it stuck at a fixed level of performance?

Execution Starter Heuristics

The execution heuristics are practical rules for parts of leadership work that are more about discipline than diagnosis, how meetings run, how priorities are set, how progress is tracked. Treat them as a starting point, not a finished system.

Running meetings

- Every meeting has one named decision or action it exists to produce.
- If there is no decision or action, do not meet.
- End with a clear owner and a clear next step.

Setting priorities

A priority is a commitment the team can finish in the time available. The test has three parts.

- **It fits.** Three priorities is the target, five is the ceiling. What matters is whether each one is achievable alongside the rest of the work, not the count.
- **It finishes.** Each priority has a measurable outcome, an owner, and a date. Without both, you have an intention.
- **It replaces.** At full capacity, new priorities only go on the list when you finish an existing one, and the new one must fit the capacity the finished one freed. Finishing a priority that took five times the effort of the others frees room for several small ones. Finishing a small one frees room for one small one. You cannot add a bigger priority than the one you just took off the list.

Tracking progress

- Make progress visible to the whole team.
- Review on a regular cadence, rather than only when something breaks.
- Adjust based on what is happening, not what was planned.

APPLY IT NOW. Write your answers below before moving on. These are starting points for a team conversation. 1. Which domain, decisions, team performance, or execution, would produce the most leverage if your team got significantly better at it? 2. Which starting set fits that domain: 5Cs, 6As, or execution heuristics? 3. What would you need to see after 30 days to know whether the starting set is working for you?

When Your Domain Crosses More Than One Starting Set

Real leadership challenges rarely stay inside a single domain. A hiring decision involves both talent judgment and decision process. A priority-setting conversation involves both execution discipline and decision clarity. When your challenge crosses sets, use this heuristic: start with the set that addresses where the pain is most visible.

If this is what you see	Start here
The problem presents as a bad outcome or poor choice	Start with the 5Cs. A clear decision process will surface the talent or execution question more precisely than starting with those domains will surface the decision question.
The problem presents as inconsistent people performance	Start with the 6As. Diagnose whether the gap is in Architecture, Alignment, Agency, Accountability, Adaptability, or Acceleration before concluding it is a decision or execution problem.
The problem presents as operational drag	Start with the execution heuristics. Running better meetings and setting clearer priorities often resolves what appears to be a talent or decision problem.
You genuinely cannot tell	Use the 5Cs to decide which set to apply first. The Clarity step, naming the real question, usually makes the right starting point obvious.

Once you have run one loop in the primary domain, the adjacent set often becomes the natural next focus. The sets are not competitors. They are sequential lenses on the same system. None of them is the method. The loop in Part One is the method.

PART THREE

Calibrate: Improve the Rules Themselves

The PACE loop improves execution of a rule. Calibrate improves the rules themselves. It is the difference between getting better at what you do and getting better at how you think about what you do.

Set a regular review, monthly at minimum, and ask these questions across the rules you have been working with:

- Which rules are producing consistent improvement? Reinforce them.
- Which rules are producing inconsistent results? Determine whether it is a rule problem, an execution problem, or a context problem.
- Which rules have outlived their usefulness? Replace them.
- What new patterns are emerging from the data that suggest a new rule worth testing?

The model is not finished. It is a living hypothesis. The goal is not to arrive at the perfect set of rules. It is to keep the improvement loop running.

Argyris and Schön's double-loop learning describes exactly this move: single-loop learning corrects errors within a system, double-loop learning periodically questions the rules of the system itself. Calibrate is where the method interrogates its own assumptions.

PART FOUR

Extend: From Individual Learning to Organizational Capability

The PACE loop builds judgment in individuals. But individual learning that stays in one person leaves with them. The method only produces organizational capability when what individuals learn transfers to others.

The simplest transfer mechanism: a periodic team-level rule review. Once a month, individuals share one thing they have learned from their loop, what worked, what did not, and what they adjusted. The group tests whether it generalizes. Does this finding hold across different people and different situations, or is it personal?

Level	What it produces
Individual loop	One person improves their judgment in one domain through iterative practice.
Team rule review	Individuals share findings. The group stress-tests them. Rules that generalize get added to the shared starting set. Rules that are personal stay personal.

Level	What it produces
Shared rule set	Over time, the team builds a validated set of rules that reflect collective learning. Not inherited best practices, not one person's opinion, but tested findings from real work.

This is how individual loops become organizational intelligence. It is also what makes the learning system resilient to turnover. The knowledge lives in the system as well as in the people.

The goal is not a team of people who learned from the leader. It is a team that knows how to learn from each other.

The Coaching Routine: Five Questions, Asked Every Week

The coaching routine is the mechanism that turns individual loops into team capability. Without it, each leader runs their own loop in isolation and the method stays personal. With it, the leader develops the next person's ability to run the loop, and the capability compounds across the team.

The five questions

- **1. What is your target condition right now?** Forces the direct report to state the outcome they are working toward, as well as the behavior they are practicing. If they cannot name a target condition, the coaching conversation stops here and restarts there.
- **2. What is the actual condition now?** Surfaces the gap between where they are and where they are trying to be. Often reveals that the direct report has been measuring effort rather than outcome.
- **3. What obstacles do you think are preventing you from reaching the target condition? Which one are you working on now?** Names the specific impediment and forces a choice. Multiple obstacles are common. Working on all of them at once is ineffective.
- **4. What is your next step? What do you expect to happen?** Makes the hypothesis explicit. The direct report is now predicting an outcome, which means the next conversation can compare prediction to reality.
- **5. When can we check what you have learned from taking that step?** Sets the next coaching meeting. Treats the step as an experiment, not a task.

Run your own loop visibly. The coaching routine only works if the person running it is also running their own loop. Direct reports will notice within weeks whether the leader practices what the

routine is teaching. If the leader asks about target conditions in coaching and has none of their own, the routine reads as something the leader imposes on subordinates rather than a shared discipline. Run your loop first. Share your target conditions openly. Model the failure diagnosis. The coaching routine compounds with the coach's own practice.

The cadence

Once a week, every direct report, fifteen to twenty minutes each. The cadence is not negotiable. Less frequent conversations lose the loop's momentum. More frequent conversations become surveillance and break the signal.

The Team Rule Review: A 30-Minute Monthly Template

Run this once a month with your direct team. The structure is fixed. The content is whatever the team brings.

Agenda, 30 minutes	Notes
Minutes 0 to 5: Each person names one rule they have been working on.	
Minutes 5 to 20: Two people share their loop in detail, what happened, what they adjusted, and what they want the group's input on. Rotate who presents each month.	
Minutes 20 to 25: Group decides which findings generalize (add to shared set) and which stay personal.	
Minutes 25 to 30: Each person commits to one rule focus for the next month.	

Governance for the Shared Rule Set

A shared rule set that no one owns becomes a graveyard. Build governance in from the start. For smaller companies, this is lighter than it sounds. For a 40-person company, the "owner" is whoever edits the shared Google Doc most often, and the "quarterly pruning" is 20 minutes at the end of an existing meeting. Scale the formality to the size of the team.

Role or practice	What it looks like
A named owner	One person, usually the team leader running the review, sometimes a designated keeper of the set, is accountable for the set's integrity. Not optional.
A pruning cadence	Each quarterly model review includes a pruning pass. Rules with no active use in 90 days are retired or archived. Conflicting rules are reconciled. The goal is fewer, better rules, not more rules.
Attribution	Each rule carries the name of the person or pattern it originated from and the date it was added. This makes it easy to check with the originator when context changes.
A retirement rule	Any rule in the set can be challenged by any team member. Challenge triggers a one-iteration test: someone runs it as a loop with explicit intent to falsify. If the rule survives, it stays. If not, it is retired. This keeps the set alive rather than accumulating.
A size cap	Soft cap of 15 active rules per domain per team. When you add a 16th, something must retire. Sets without a cap become shelfware.
Turnover protocol	When the originator of a rule leaves the team or company, the owner tags the rule for review at the next quarterly. Unclear rules from departed originators tend to drift. Explicit review prevents that.

What to Track: A Measurement Starter Kit

You do not need a dashboard. You do need enough signal to know whether the method is working and whether to adjust. This section provides that signal at three cadences.

Process indicators (weekly, per leader)

- Loop completed: yes or no.
- External signal collected: yes or no.
- Adjustment identified: yes or no.

Two minutes to track. The discipline is itself a signal. If loops are not being completed, that is the first thing to fix, not a measurement problem.

Controls, aka leading indicators (monthly, by starting set)

Indicators specific to the set you are running. These are what you watch for trend change in the first 90 days.

Starting set	Controls to watch
5Cs (decision quality)	Percent of decisions where assumptions were explicitly named before options were discussed. Percent of decisions with an explicit owner and next step at the moment they are made. Number of decisions that get re-litigated in the following two weeks.
6As (team performance)	For the dimension you are working on: a short check-in on that dimension with your team, taken monthly. For Agency: number of escalations the leader received that their report could have handled. For Alignment: whether direct reports can articulate the top three priorities unprompted.
Execution heuristics	Percent of meetings that ended with a named owner and next step. Number of active priorities per leader (target: three to five). Ratio of work stopped to work started in the last 30 days (target: at least one stopped for every one started).

Consequences, aka lagging indicators (quarterly, outcome level)

These are what you eventually expect to see move if PACE is working. They will lag the controls by 60 to 120 days.

- Decision cycle time for the kinds of decisions the team works on.
- Quality of decisions, measured by re-litigation rate and by board or customer feedback when applicable.
- Escalation rate to the leader for decisions that should be made below them.
- Number of rules in the shared set that are in active use by multiple people.
- On the 6As check-in, movement on the dimensions you have been working on.
- Psychological safety score, re-measured. Should trend up or stay steady. A decline is a warning.

You are not measuring effort. You are measuring whether judgment is improving and whether the organization is becoming less dependent on any single person to make good decisions.

Closing the Loop: What This Produces Over Time

A CEO who runs PACE for a year does not end up with a bigger rule set. They end up with a different kind of organization.

Decisions that used to come back to them twice now get made once and stay made. The escalation rate drops, and their calendar has room in it again. The leadership team argues better, which means faster, with fewer assumptions treated as facts. New hires can look at the shared rule set and see how this company thinks, which means ramp time shortens. When someone leaves, what they learned does not leave with them. The 6As check-in moves in the direction you chose to move it. The psychological safety score holds or climbs.

None of this is visible in month one. By month six a careful observer would see it. By month twelve the team knows it and so do their customers.

The goal is not a team of people who learned from you. It is a team that knows how to learn from each other, and from the work, faster than the environment is changing them. That is what adaptive capability looks like.

FULL IMPLEMENTATION

Installing PACE in Your Leadership Team

You have read the method. You have either run the Getting Started Guide yourself or you are ready to start the cycle and the rollout at the same time. This section is the staged rollout that turns personal practice into team capability.

Do not launch organization-wide on day one. Stage the rollout. The leader who has not personally practiced the method cannot install it well in others, and the team that watches a leader skip steps will skip them too.

The Staging

Phase	What happens
First leadership team	Bring your direct reports into the cycle. They run their own loops. You coach them through the routine in Part Four. You hold the first team rule review at week four. Plan four to six weeks for this phase.

Second team	A peer of yours brings their team in. Your team has enough live experience to coach theirs. The first team's shared rule set becomes a starting reference for the second.
Beyond	Add teams whose leaders are ready. Ready means the leader has run loops personally for at least four weeks, is not in active crisis, and has a team with working psychological safety. Leaders who do not meet the bar wait for the next wave.
Steady state	After the rollout, the cadence is one cycle per week, one team rule review per month, one model review per quarter. That rhythm is sustainable and compounds.

If you are under 75 people, you may stop after the first leadership team and never need the rest. Your shared rule set lives in a Google Doc that one person on the team keeps updated. Skip the second team phase entirely. The staged rollout matters more when you have multiple intact teams to sequence.

If You Are Over 300 People

The method does not change. The rollout does. Three things shift when the organization is large enough that the CEO can no longer personally run the cycles for the whole leadership team.

Your role becomes sponsor, not runner

You still run your own cycle. You still coach your direct reports. But you stop running cycles for everyone else. Your job becomes making sure the method has air cover at the top, the resources to install it well, and consequences for leaders who claim to be running it but are not. Without that air cover, the method is one more initiative competing for attention. With it, the method becomes the way the company learns.

Pick one division and prove it works

Do not roll out across divisions on day one. Pick one division where the leader has already run the cycle personally, has a team with working psychological safety, and is operating in conditions where PACE will produce visible signal in 90 days. Let that division become the proof and the source of internal exemplars for the rest of the company. The leader who runs the first division becomes the coach for the second.

Replace leadership team with division in the staging

The phases above still apply, scaled up. "First leadership team" becomes "first division." "Second team" becomes "second division." The staging from Phase 1 to Phase 2 takes a quarter or longer rather than four to six weeks. Steady state cadence at the division level matches what the leadership team was doing earlier: weekly cycles, monthly team rule reviews, quarterly model reviews.

Governance earns its keep at scale

The shared rule set governance described in Part Four stops being optional. You need a named owner for each domain's rule set, a quarterly pruning cadence that happens on schedule, and a retirement test that any team member can trigger. At 300 people the shared set has enough mass to drift if no one is watching it. The governance is what keeps it from becoming shelfware.

Watch for divisions that go quiet

In a multi-division rollout, the failure mode is not active resistance. It is silence. A division that stops reporting on its loops, stops adding to the shared set, and stops surfacing rules in the quarterly model review is a division that has either dropped the method or is hiding that it has. Both look the same from the outside. The CEO's job as sponsor is to notice the silence and ask what happened, before the rollout fails by attrition rather than by decision.

EXAMPLES

The Method in Practice

These examples are drawn from coaching engagements. Read them for how the loop diagnoses, adjusts, and recovers.

Example 1: Improving Decision Quality

A leadership team at a mid-sized company notices that their weekly decision meetings are running long and producing weak alignment. Decisions made in the meeting frequently get relitigated the following week.

They apply the 5Cs as their starting set. The leader focuses on one rule: separate facts from assumptions before generating options. After three meetings, self-assessment shows the team is still mixing the two. The leader is not stopping the discussion early enough. External signal from two team members confirms: the discussion feels circular because everyone is working from different underlying beliefs about what is true.

Diagnosis: wrong execution, not wrong rule. Adjustment: at the next meeting, the leader pauses at the start of the options phase and explicitly names the assumptions on the table before allowing anyone to advocate for a direction. After five iterations, decision time drops and re-litigation drops with it. The team adds a new rule: every decision discussion begins with a one-minute explicit inventory of what is known versus what is assumed.

Loop summary	
Rule used	Separate facts from assumptions before discussing options (5Cs: Clarity).
Failure diagnosis	Wrong execution. The rule was sound, the behavior was inconsistent.
Adjustment	Pause at the start of the options phase and explicitly name assumptions before advocacy begins.
Outcome	Decision time reduced, re-litigation dropped, team added a new shared rule.

Example 2: Building Team Agency

A CEO of a 60-person company recognizes that her team brings her too many decisions they should be making themselves. She diagnoses it as an Agency problem using the 6As. Rule: ask before telling. In every one-on-one and team discussion, ask the other person what they think should happen before offering her own view.

After two weeks, self-assessment shows she is doing it in one-on-ones but defaulting to telling in group settings. External signal confirms: in groups, people are still deferring to her because she speaks first. Diagnosis: wrong execution in the group context.

Adjustment: in every group setting, she commits to speaking last. She adds a second rule: state the decision boundary clearly at the start. "This is yours to decide, I am here to help you think." Over three months, escalations to her decrease meaningfully. The quality of decisions made without her improves, measured by how often those decisions need revisiting.

Loop summary	
Rule used	Ask before telling; speak last in group settings (6As: Agency).
Failure diagnosis	Wrong execution in group context. Rule was right, behavior was inconsistent.
Adjustment	Commit to speaking last; open with an explicit decision boundary statement.
Outcome	Escalations to CEO decreased meaningfully; decision quality without her improved.

Example 3: When the First Diagnosis Is Wrong

The first two examples follow a clean arc. Real loops rarely do.

A VP of Sales at a 120-person company is trying to improve forecasting accuracy. Deals are consistently closing later than forecast, creating cash flow pressure and credibility problems with the board.

She identifies a rule from the 5Cs to focus on: create choices. Her hypothesis is that her team is anchoring on a single scenario and not stress-testing it. She runs the loop for three weeks, coaching the team to surface at least three forecast scenarios in weekly reviews. Outcomes do not improve.

She diagnoses the failure as a wrong rule and replaces it with a new rule focused on the Clarity step, defining the decision being made in each forecast conversation more precisely. She runs that for two weeks. Still no improvement. She has now changed rules twice without a positive result.

Before changing a third time, she goes back to the external signal step and asks differently. Not about the forecast conversations, but about what the team believes is causing deals to close late. Three separate team members, unprompted, describe the same pattern: customers are not being asked for explicit commitment at the right moment. The conversation is clear. The choices are surfaced. But no one is closing the loop.

Her original failure diagnosis was wrong. It was not a decision framework problem. It was an execution problem on a rule she had never made explicit: end every forecast conversation with a direct question about the customer's next decision point.

Within four weeks of applying that rule consistently, forecasting accuracy improves materially.

The external signal step, asked with genuine curiosity rather than confirmation-seeking, surfaced what self-assessment had missed. The loop corrected itself.

Loop summary	
Rule used	End every forecast conversation with a direct question about the customer's next decision point.
Failure diagnosis	Initial misdiagnosis (wrong rule). Actual problem was wrong execution of an unstated rule.
Key recovery moment	Asking the team what they believe causes the problem, not asking them to evaluate the leader's performance.
Outcome	Forecasting accuracy improved materially within four weeks of applying the correct rule consistently.

Example 4: A Team Running PACE Together

The first three examples show individual leaders running loops. The fourth shows what happens when the coaching routine and the team rule review are running alongside the loops, and how that combination produces capability at the team level rather than only the individual level.

A CEO of a ninety-person company is trying to reduce escalations to her and her leadership team. Her six direct reports are bringing too many decisions up rather than handling them at their level. She starts with the coaching routine. Every Monday, in each direct report's one-on-one, she runs the five questions in order. The first week is awkward for everyone. Two direct reports describe weekly tasks rather than target conditions. She does not correct them. She asks the next question, then the next, and at the end of each conversation asks what target condition they will bring to next week's. By the fourth week, all six arrive with a named target condition and an honest account of the gap.

In parallel, she launches the monthly team rule review. Two direct reports present each month. Some rules get promoted into the shared set; others stay personal when the group cannot confirm they generalize. Over the following months, escalations drop, the leadership team argues better at their level, and rules the CEO developed in her own loop get translated rather than transplanted by her direct reports. The coaching routine is what made the individual loops compound. The team rule review is what made the rules collective. Neither alone would have produced the result.

Loop summary	
Individual loops running	Six direct reports, each on a rule they chose from the 5Cs or 6As, each with a target condition.
Coaching routine	Weekly, five questions, fifteen to twenty minutes per direct report.
Team rule review	Monthly, two people present in depth, group decides what generalizes.
Result at six months	Escalations down roughly 40 percent, shared rule set of 9 active rules used across the leadership team, new-hire ramp time shortened by visible use of the shared set.

ADVANCED APPLICATION

Using PACE When the Environment Is Wicked

PACE was designed for the work that occupies most of a leader's time. But not all leadership challenges are the same kind of hard.

Complicated: the right answer is discoverable through analysis. Feedback is reasonably clear. Experience transfers. Hiring a strong candidate, running a disciplined budget review, improving how decisions get made. Hard, but solvable with good thinking and good process. The method works as written.

Wicked: the right answer is only visible in retrospect, if at all. Entering a new market. Rebuilding culture after a leadership failure. Making a strategic bet on a technology that does not yet exist. Feedback is slow, partial, or misleading. Experience can deceive as much as guide.

Most CEOs operate in both simultaneously, complicated work running inside a wicked context. The distinction matters because the method needs to be adapted for which environment the specific challenge lives in.

Environment	What it means and when you are in it
Complicated	You can figure out the right answer by analyzing the situation carefully. Feedback is reasonably clear and timely. Experience transfers reliably. Examples: improving how decisions get made, hiring disciplines, meeting effectiveness, priority setting. The method works as written.
Wicked	The right answer is only visible in retrospect, if at all. Feedback is slow, partial, or misleading. What worked before may not work now. Examples: entering a new market, rebuilding culture, making strategic bets under genuine uncertainty. The method adapts, as described in the rest of this section.

*The terms kind and wicked learning environments come from Robin Hogarth's research on expertise, later developed into Dave Snowden's **Cynefin** framework. In complicated domains, cause and effect are discoverable through analysis. In complex domains, cause and effect are only visible in retrospect. The system changes as you interact with it.*

Step Zero in a Wicked Environment

Step Zero does not disappear. It becomes more important. What it is calibrating for changes fundamentally. In wicked environments you are not looking for experts with proven track records. You are looking for people with informed hypotheses and the intellectual honesty to know the difference between what they have tested and what they believe. Rules presented as proven answers in complex domains are a warning sign. Add Tier 4, adjacent domains, people who have navigated analogous complexity in different fields. Cross-domain pattern recognition is often more reliable than domain expertise.

How the Loop Changes in a Wicked Environment

Loop step	How it adapts
Step 1: Intent	Still choose one rule, but expect to hold it longer before evaluating. A loop that takes a week in a complicated environment may take a quarter in a wicked one. Set your intent with that timeline in mind.
Step 2: Action	Instead of applying one rule to one situation, run what Snowden calls safe-to-fail probes: small, bounded experiments designed to generate signal without betting the organization on the outcome. Two or three in parallel, in different contexts. The question for each is not “did this work?” but “what did this reveal?” Comparing what different probes reveal is where the learning lives.
Step 3: Self-Assessment	Add a pre-mortem question: if this rule fails over the next 90 days, what is the most likely reason? This surfaces hidden assumptions before the feedback arrives, not after.
Step 4: External Signal	Look for weak signals, early peripheral indicators before the outcome is visible. Ask different questions. What are you noticing that seems small but might matter? What are people not saying that they probably should be?
Step 5: Adjustment	Adjustments are tentative. Treat them as hypotheses to be tested, not corrections to be implemented. Run multiple small adjustments in parallel rather than betting on one.
Step 6: Repeat	Iteration speed matters less than iteration quality. Rushing the loop produces false signal. Slow down the cadence but maintain the discipline.

The Pre-Mortem

In complicated environments, feedback tells you whether a rule worked. In wicked environments, you often cannot wait for feedback. The stakes are too high and the signal too slow. The pre-mortem is a substitute.

Before applying a rule in a high-stakes, slow-feedback situation, spend ten minutes answering this question in writing: it is six months from now and this approach has clearly failed. What happened?

Work backwards from that imagined failure to identify the assumptions your rule depends on that might not hold.

This does two things. It surfaces hidden context-dependence, the conditions under which your rule breaks, before you discover them the hard way. And it generates the weak signal questions you should be asking during the loop. If assumption X is wrong, what early evidence would I see?

The pre-mortem was developed by psychologist Gary Klein and validated in research on decision quality. It works because imagining failure is cognitively easier than predicting it. The brain generates more and better hypotheses when it starts from a failure state than when it starts from uncertainty.

Safe-to-Fail Probes

There are two forms of parallel probing. The first, running the same rule across different contexts and with different people simultaneously, is described in Step 2. The second, running different candidate rules in parallel before committing to any single one, is what this section addresses. It is most useful when you are still in Step Zero and have genuine uncertainty about which rules are worth testing at all.

In complex environments, Snowden's Cynefin framework prescribes running multiple small probes simultaneously rather than implementing a single solution. The logic is that in a system where cause and effect are unclear, you need to generate signal from several directions at once, then amplify what seems to be working and dampen what is not.

For a CEO, this translates as follows: instead of picking one rule and running it for a quarter, identify two or three candidate rules for the same domain and run them in parallel, in different contexts, with different team members, in different kinds of situations. Compare what you learn across the probes before committing to any single rule.

This is a deliberate departure from the one-rule-at-a-time discipline of the standard loop. That discipline exists to manage cognitive load and focus. In wicked environments the cost of committing to the wrong rule early is higher than the cost of the added complexity.

Common Ways This Breaks

Common failure modes. The two most dangerous, weak psychological safety and leader reintroducing dependency, are covered in depth in Part One.

Failure mode	What to do about it
Weak psychological safety	Stop running PACE. Produces false signal that pollutes the shared set. Address safety first. See the check-in in the Getting Started Guide.
Leader reintroducing dependency	The single biggest killer of PACE in adopting teams. See the dedicated section in Part One for tells and antidotes.
Coaching routine collapses into problem-solving	The leader answers the five questions instead of asking them. Direct reports learn the real answers come from the leader. The routine becomes a disguised directive. Reset by running the five questions in order, in silence when necessary, for at least a month.
No target condition	Loops run indefinitely with no falsification test. Leaders feel effort but cannot evaluate whether effort is producing outcome. Set a target condition before the loop begins, every time.
Too many rules at once	Cognitive load prevents real focus. Stick to one rule per iteration.
Vague rules	A rule you cannot observe cannot be practiced or improved. Make rules behavioral and specific.
No external signal	Self-assessment alone has blind spots. The external signal step is not optional.
Weak source calibration	Building from the wrong expert produces confident mediocrity. Do the Step Zero work.
Treating the starting set as doctrine	If the 5Cs or 6As are never questioned, they become a playbook. Treat them as hypotheses.
Using popularity instead of outlier logic	What most organizations do is not the same as what the best organizations do. Build from outliers.
Unmanaged shared rule set	Without an owner, a pruning cadence, and a size cap, the set becomes shelfware. See Governance for the Shared Rule Set.

RESEARCH FOUNDATION

Where the Claims Come From

This appendix traces each claim in the toolkit to its source. It is written for the reader who wants to know whether the method rests on evidence or intuition. The short answer is evidence. The longer answer is below.

A Candid Note on Evidence

PACE synthesizes well-established research into a working method. The AlphaGo Zero principle, deliberate practice, double-loop learning, psychological safety, Cynefin, the pre-mortem, and the neuroscience of learning consolidation are all supported by decades of published work. What follows traces each claim to its source.

Treat PACE as a calibrated hypothesis drawn from sound research: worth running, worth measuring, and worth revising as evidence accumulates in your context.

Step Zero: Calibration

The validation logic in Step Zero draws on two bodies of work. The first is deliberate practice research, pioneered by Anders Ericsson and summarized in *Peak* (2016), which establishes that expert performance is not a function of experience alone but of the quality and structure of practice, and that not all experts have developed transferable models of their expertise.

The second is signal detection theory and the logic of using multiple sources to separate signal from noise. Daniel Kahneman's *Noise* (2021, with Sibony and Sunstein) documents how much random variation exists in expert judgment, even among highly credentialed professionals, and makes the case for structured validation of rules rather than deference to authority. The rule gauntlet in this toolkit is a direct application of that logic.

The PACE Loop

The six-step PACE loop is grounded in several converging research streams.

Deliberate practice (Ericsson): improvement requires focused repetition on specific components with immediate feedback, not general experience. The loop's structure, one rule, real action, specific reflection, external signal, one adjustment, mirrors the conditions Ericsson identifies as necessary for skill development.

Double-loop learning (Chris Argyris and Donald Schön, *Organizational Learning*, 1978): effective organizational learning requires correcting errors within a system as well as periodically questioning the rules of the system itself. The Calibrate phase of PACE is a direct implementation of double-loop logic.

The AGES model (NeuroLeadership Institute, David Rock and colleagues): research on how the brain consolidates learning identifies four conditions that drive retention and transfer: Attention, Generation, Emotion, and Spacing. The loop creates all four. Focused attention on one rule, generation through real-world application, emotional engagement through actual stakes, and spacing through repeated iterations over time.

John Medina's Brain Rules (2008) contributes the spacing and repetition findings: the brain does not retain information from single exposures. Repeated retrieval and application across time is required for durable learning. The loop's insistence on fast, repeated iteration is neurologically grounded.

Adult Learning

The andragogical premises articulated by Malcolm Knowles, that adults learn best when learning is self-directed, connected to immediate problems, and applied in real contexts, are foundational to the PACE design. PACE refuses to separate learning from work because Knowles' research and 50 years of subsequent adult learning literature establish that the separation itself is the source of most failed leadership development.

Psychological Safety and the External Signal Step

The external signal step assumes that honest feedback is available. Amy Edmondson's research on psychological safety (The Fearless Organization, 2018, and decades of prior work) establishes that teams with low psychological safety systematically suppress the honest signal that learning loops require. Her finding, that psychological safety is the top predictor of team learning and the top predictor of team effectiveness in Google's Project Aristotle research, is the scientific basis for treating safety not as a cultural nicety but as a structural prerequisite for this method to function. The psychological safety check-in in the Getting Started Guide is a direct operationalization of this research.

The method's instruction to ask for experience rather than evaluation ("what was your experience of that conversation?" rather than "how did I do?") is grounded in Edmondson's guidance on how to solicit useful signal without triggering defensiveness or social desirability bias.

The 5Cs Decision Framework

The 5Cs framework is supported by Kahneman's System 1 and System 2 research (Thinking, Fast and Slow, 2011), which establishes that most decision errors are not failures of intelligence but failures of process. Specifically, the failure to slow down pattern-matching and surface hidden assumptions. The Context and Clarity phases of the 5Cs are direct implementations of the deliberate System 2 activation Kahneman identifies as necessary for high-stakes decisions.

The Create Choices component addresses what Kahneman and colleagues call narrow framing, the tendency to present decisions as binary when more options exist. Research consistently shows that decision quality improves when the choice set is deliberately expanded before evaluation begins.

The 6As Team Framework

The 6As framework draws on several research foundations. Edmondson's teaming research establishes that high-performing groups share mutual understanding of goals, roles, and norms, the conditions that Architecture and Alignment are designed to create. Her finding that alignment predicts execution velocity is replicated across multiple organizational studies.

David Rock's SCARF model (Status, Certainty, Autonomy, Relatedness, Fairness) provides the neuroscience basis for the Agency dimension. Rock's research shows that autonomy, the sense that one can act without waiting for permission, is one of the primary drivers of motivated behavior. The Agency dimension of the 6As is designed to create the structural conditions SCARF identifies as necessary for people to function at full capacity.

W. Edwards Deming's foundational insight, that the majority of performance problems are system problems, not people problems, underlies the entire 6As diagnostic logic. The 6As are a framework for examining the system before attributing performance failures to individuals.

Target Conditions and the Coaching Routine

The explicit naming of target conditions and the structure of the coaching routine come from Mike Rother's research on how Toyota develops its managers, documented in *Toyota Kata* (2010) and subsequent work. Rother distinguishes two routines: the Improvement Kata, which is the learner's pattern of working toward a target condition through experiments, and the Coaching Kata, which is the structured conversation a coach has with the learner to develop their ability to run the Improvement Kata independently.

The target condition is the specific difference Rother's work surfaces between Toyota's approach and Western continuous improvement practice. A standard continuous improvement program focuses on solving current problems. Toyota's Improvement Kata focuses on closing the gap between current state and a named target condition, which reframes the work as experimentation toward a specific outcome rather than a list of fixes.

The five-question coaching routine in Part Four is adapted from the Coaching Kata Rother documents. The original is designed for manufacturing contexts. The executive adaptation changes the language and the cadence but preserves the structural logic: consistent questions in a consistent order, asked frequently enough to build the learner's pattern of thinking.

Rules as Hypotheses

The framing of organizational rules as testable hypotheses rather than commandments comes from two traditions. The first is the scientific method itself, applied to organizational practice through authors like Stefan Thomke (*Experimentation Works*, 2020), who argues that organizations produce better outcomes when they treat their operating assumptions as hypotheses to test rather than truths to follow.

The second is the TPS tradition's treatment of standards. A standard at Toyota is not a correct answer that has been discovered. It is the current best known approach, held explicitly so that deviation produces visible signal. Without a standard, drift is invisible. The standard exists to make learning possible, not to enforce compliance. This framing is developed across Taiichi Ohno's *Toyota Production System* (1988), Shigeo Shingo's work, and Jeffrey Liker's *The Toyota Way* (2004).

Structured Failure Diagnosis

The four-question diagnostic structure in Part One is a compressed adaptation of A3 thinking, a problem-solving discipline from the TPS tradition documented in John Shook's *Managing to Learn* (2008) and Durward Sobek and Art Smalley's *Understanding A3 Thinking* (2008). Full A3 thinking uses a single page to document background, current condition, goal, analysis, proposed countermeasures, and follow-up. The executive version in PACE preserves the core discipline of naming alternative explanations before committing to a diagnosis, without requiring the full document.

Wicked Environments, Cynefin, and the Pre-Mortem

The distinction between kind and wicked learning environments originates in Robin Hogarth's research on expertise, documented in *Educating Intuition* (2001). Hogarth found that expertise develops reliably in kind environments, where feedback is clear, timely, and accurate, but can actively mislead in wicked ones, where feedback is delayed, ambiguous, or absent. This finding is the scientific basis for the toolkit's guidance that standard loop cadence needs to slow down in complex situations.

Dave Snowden's Cynefin framework (developed at IBM in the late 1990s and refined extensively since) provides the most widely used practitioner model for distinguishing between complicated and complex domains. In complicated domains, cause and effect are discoverable through analysis. Experts can identify right answers. In complex domains, cause and effect are only visible in retrospect. The system changes as you interact with it and right answers emerge rather than being designed. Snowden's prescription for complex domains is safe-to-fail probes: multiple small experiments run in parallel to generate signal, with amplification of what works and dampening of what does not. The wicked environment section of this toolkit is a direct application of that logic. Snowden's foundational paper is "A Leader's Framework for Decision Making" (*Harvard Business Review*, November 2007, with Mary E. Boone).

The pre-mortem technique is documented in Gary Klein's research on naturalistic decision making, summarized in *The Power of Intuition* (2003) and *Sources of Power* (1998). Klein found that prospective hindsight, imagining that an event has already occurred and working backwards, significantly improves

the ability to identify potential failure points compared to prospective analysis. Daniel Kahneman subsequently incorporated the pre-mortem into his decision quality framework as a structured debiasing technique, recommending it specifically for high-stakes, low-feedback decisions of exactly the kind that characterize wicked environments.

The Performance Operating System Series

catalystgrowthadvisors.com