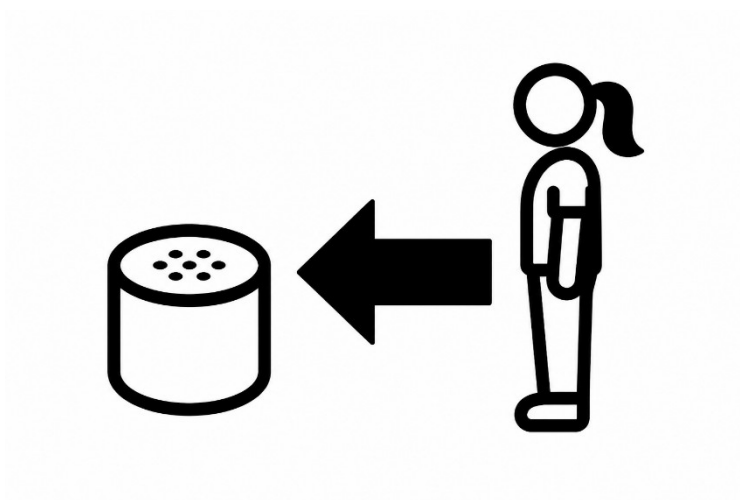


Kindel Leadership Development



Product Leadership

Introduction.....3

Introduction

Product Leadership is Leadership

At the 1996 Microsoft Professional Developer Conference (PDC) I stood up in front of 8,000 customers and announced what I had been working on for the previous two years: the Distributed Component Object Model (DCOM). I wrote and demoed code on stage, and every attendee went home with a CD-ROM holding a beta version to try.

A few weeks later, one of those 8,000 customers (yes, only one) called Microsoft Support to point out that the most critical file required to make DCOM work was missing from the CD-ROM.

We literally gave AMAZING TECHNOLOGY away to 8,000 developers, and only one cared. The engineering was not the problem. We had started with a technology and then gone looking for ways customers might use it.

I vowed then and there to never again build something without starting with the customer. Making that vow real is a leadership job, and in most companies, it lands on the people who lead product.

We call those people product managers, but manage is the wrong term. The right term is lead. We should call them Product Leaders. A PM who makes a customer problem so clear that the engineers invent a better answer than the roadmap imagined: that person is leading. A PM manager who teaches a new PM to say no and explain why: also leading. A head of product who keeps marketing, sales, support, and supply messaging consistent with what the builders built: leading. A builder who reads a working backwards doc, spots the hole in the customer experience, and says so in the review: leading.

This course is about becoming a better leader. The skills in this workbook apply whether you are an engineer with no dedicated product management support, are new to the discipline, or lead an entire product organization. They apply whether your title says “Software Design Engineer,” “Product Manager,” “Director of Product,” or “Chief Product Officer.” They apply because product leadership is a craft, not a job title, and craft improves only through deliberate practice.

Thirty Years of Building Products (Well and Poorly)

I have been leading teams and building products for over thirty years. Some of those years were spent at enormous scale: Microsoft, where I helped ship Windows, Internet Explorer, and Windows Phone to hundreds of millions of customers; Amazon, where I built the Alexa Smart Home organization from 7 people to 150 in under a year and helped define how voice-first computing would work in people's homes. At Control4/SnapOne I was CTO of a 4,500-SKU smart home company navigating a merger and a public-to-private transition. I have also founded startups including FreeBusy, Holy Grail Labs, and Excaliwire, where building from zero means every decision is a one-way door with real consequences.

My first real chance to lead brand-new product development came in 1999, when I helped found the Microsoft Connected Home Business Unit, which ended up building Windows Media Center. I invented my own version of working backwards: the team mocked up the “back of the box” and wrote the review article we wanted our favorite PC Magazine reviewer to write, long before we selected a technology.

Across all of these (big and small, and successful and not), I have learned one thing with certainty: the skills that make product leaders effective do not change with scale. What changes is the cost of getting them wrong. At 20 people, a bad decision about priorities wastes a sprint. At 2,000 people, that same bad decision wastes a quarter and erodes trust across an organization. The principles are the same. The stakes are not.

I have also learned that no one gets it right all the time. I have been on the team that built a technology nobody wanted. I have been the exec who “randomized” his teams with half-baked ideas on a Monday morning. I have been the person who confused being busy with being effective. Every failure taught me something; usually that I had relied on good intentions when what I needed was a mechanism, a principle, or the courage to say no.

This workbook is the distillation of those lessons as they apply to product: the ones I learned by doing, the ones I learned by failing, and the ones I learned by watching other great product leaders operate; often in ways I initially disagreed with, until the results proved me wrong.

The Essence of Product Leadership

Here is a truth that makes some people uncomfortable:

Leadership is getting others to do what you believe is right; and having them believe it was their idea.

Read that again. If you are a PM, it should read like a job description. A PM rarely has authority over anyone. The engineers do not report to the PM. Neither do the designers, the marketers, the sellers, support, or supply chain. Product Management is an influence job, and being a PM means leading people whose every result arrives through someone else's hands.

The mental model I teach is simple to say and hard to live:

Builders create the product. Every other function creates (or destroys) the experience of that product in the world. The PM's job is to create the conditions for magic across the whole chain, in a way that honors and reinforces one coherent customer experience.

Builders are the humans and AI agents who design, implement, and ship the product. That is not a status ranking but a value-creation fact. The PM is not the builder of record, does not own the engineers, and does not win by producing documents. The PM wins by creating clarity of direction, forcing real tradeoffs, holding a high bar for customer obsession when the customer cannot be in the room, and enabling every other function to do its magic without rewriting the product by accident.

Most organizations build an insulation model instead: customer to PM, PM to (sometimes) a product owner, product owner to engineering. Every handoff adds fog. Builders get farther from the problem, and the experience fractures somewhere between "we shipped" and "the customer got what we promised." That model assumes builders cannot understand customers or business problems. Wrong.

So, what does a PM own? Not the product, and not the builders. A PM owns the system that produces good product decisions when they are not in the room:

- The definition of the PM job, written down and shared, so PMs, builders, and executives mean the same thing by it.
- The mechanisms PMs use to create clarity, starting with working backwards reviews and written tradeoffs.
- The customer experience standard that every function touching the product is held to.
- The culture that rewards all of this when nobody is watching.

AI agents make this system more important, not less. When agents do much of the making, change volume rises and shared context gets scarce. PM craft does not shrink; it moves up to outcomes, discovery, delightful experiences, tradeoffs, influence, and shared meaning, and to conditions that stay true while builders (including agents) thrash the middle. Humans still own judgment.

Leading Without Authority

The most important form of influence is the kind that does not rely on authority at all.

When anyone tries to drive change using only their title, it is rarely successful in the short term and never in the long term. Most PMs do not even have that option. "Because I said so" may work on a kindergartner. It does not inspire confidence or lasting results in a room of builders who know more about the system than you do.

I also teach Leaders of Leaders Training, which covers the behaviors that earn influence without a title: diving deep, earning trust, thinking big, holding high standards, writing it down, building coalitions, and making it “their” idea. In many ways, this course applies those same behaviors in the three directions a PM must influence every day:

- PMs influence builders with context. Influencing builders is not about compliance; the goal is ownership.
- PMs influence up with clarity. Executives usually want judgment, implications, risk, options, and tradeoffs, not all the detail.
- PMs influence out with shared purpose and a non-negotiable customer experience standard. Sales, marketing, support, supply, quality, and regulatory each bring different incentives, lexicon, and definitions of success.

The difference shows up in the first sentence of a request: a PM who says “Build this” gets compliance at best. But a PM who says “Here is the customer problem. Here is why it matters. Here is what success looks like. Here are the constraints. Here are the tradeoffs we understand so far. Help me find the best answer” gets ownership. Oh, and it turns out the same brief works for agents: It is the difference between useful output and confident wrongness (aka slop).

The Secret: Be Principled

If influence is the engine of leadership, principles are the steering. Without them, influence becomes self-serving. With them, influence becomes something people trust, follow, and eventually replicate.

In over thirty years of building products and organizations, I have concluded that most leadership books are great examples of survivorship bias. None of them clued me into the real secret of what distinguishes teams that consistently deliver outsized results from teams that are just mediocre.

The secret is principles.

Principled leaders (ahem, PMs) have a set of strongly held beliefs about the how (not just the what, why, when, or whom), and they strive to live those principles every day. Even better, organizations that have a written-down set of rules for the how, combined with mechanisms that reinforce the living of those principles, sing.

In product work, principles show up at two levels. The first is personal: mine has been “start with the customer, not the technology” since that CD-ROM shipped without its most important file. The second belongs to the endeavor. Every working backwards doc answers a set of required FAQs, and one of them is “What are our Principles for this product/program/feature?” Tenets written for a specific product are durable tiebreakers, and they let a team decide the hard tradeoffs before it is in the meeting where everyone is tired.

This workbook is grounded in a few core beliefs about product:

- Builders create the product. PMs create the conditions for magic.
- Influence is the job. Principles keep it honest.
- Start with the customer, not the technology. Clarity of the customer experience comes before deciding what to invent.
- A roadmap is not strategy. Strategy is the set of choices that focuses the organization on the right outcomes.
- Solving the pain is the floor. Delight comes from how little the product fights the customer.
- PMs are not paid to make every stakeholder happy. They are paid to help the organization make better decisions.
- Alignment does not happen when people use the same words; it happens when they attach the same meaning to those words.
- Hero PMs do not scale. Mechanisms, cadence, shared meaning, and compounding standards do.

Everything that follows rests on that foundation. Each chapter is a specific skill or system for translating those beliefs into daily product work at scale. In teaching order:

1. Builders Create the Magic; PMs Provide the Environment establishes the shared mental model and lexicon.
2. Influence Without Authority applies principled influence to the PM’s daily work.
3. Prioritization and Tradeoffs teaches decision-making under constraint, including saying no.
4. Influencing Builders (Human and AI) trades compliance for ownership.
5. Influencing Up and Out brings clarity to executives and one experience standard to every other function.

6. Shared Meaning, Not Shared Words replaces false alignment with explicit definitions and decisions.
7. Working Backwards from the Customer installs customer-before-technology as a tenet and teaches the mechanism, from the required FAQs to running a series of reviews.
8. Product Outcomes and Product Strategy moves PMs from activity to outcomes, and from roadmaps to choices.
9. Crafting Delightful Customer Experiences treats delight as a requirement, not a polish pass.
10. Teaching (human PM craft) scales product excellence by building PMs who can create more PMs.
11. Building Product Culture turns individual skill into a system.

What This Course Is (and Is Not)

At small scale, product can succeed through proximity, heroics, and one great PM who holds everything in their head. At scale, those fail. What replaces them is principles, mechanisms, cadence, shared meaning, and discipline. PMs do not personally make every product call; they build PMs and systems that make good calls repeatedly.

It is worth being clear about what this course is not.

- It is not a Jira or roadmap class. It will not teach you how to configure a tool, run sprint ceremonies, or groom a backlog. Those things matter. They are not this.
- It is not a course in being a better translation layer. The PM who insulates builders from customers is the problem this course exists to fix.
- It is not a collection of templates to admire. Working Backwards is a mechanism, not a template, and every chapter ends in practice on a live product, because a model you have not applied to your own product is trivia.
- It is not neutral. This is one practitioner's opinionated take, formed over thirty years and stated plainly enough to argue with. Where you disagree, write down why. That disagreement is itself a principle worth debating.

By the end of this course, you will have practiced specific skills, built specific artifacts (a working backwards doc reviewed at least twice, a strategy one-pager, a no-list, a tradeoff memo, and a lexicon), and committed to specific actions; with dates, with owners, and with the principles to guide you when the path gets hard.

How to Use This Workbook

This is a workbook, not a book. Every chapter has the same spine: the goals, the topics, the key lesson, and the practice. The prose exists to make the practice make sense. Practice is where the work happens.

Three things to know before you start.

- Bring a live product. The practice asks you to map, rewrite, and review real work: a real strategy, a real working backwards doc, and a real tradeoff. Vague answers produce vague results, so use the actual product, the actual names, and the actual dates.
- Chapters stand alone, but the order is deliberate. Builders Create the Magic always comes first, because every later chapter assumes its mental model and its lexicon. If your room already uses the same words to mean different things, pull Shared Meaning, Not Shared Words forward.
- Some practice needs other functions in the room. Later working backwards reviews need the people who will build and sell the product, and you cannot reconcile what "customer" means to a builder, a seller, and support without all three.

The material works self-paced, as a facilitated cohort, or as the spine of a product leadership offsite. In a workshop, each chapter runs about a half day. Take what your situation needs and leave the rest. Just do not skip writing.

Let's get to work.

The rest of this workbook is available on request.