PROJECT MANAGEMENT

From Initiation to Closure



RADIM KAUFMANN

PROJECT MANAGEMENT: From Initiation to Closure A Practitioner's Complete Handbook

Copyright © 2026 Radim Kaufmann. All rights reserved.

First Edition, 2026

No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other non-commercial uses permitted by copyright law.

For permissions and inquiries: www.radimkaufmann.com

PMBOK is a registered trademark of the Project Management Institute, Inc. PRINCE2 is a registered trademark of AXELOS Limited. All other trademarks are the property of their respective owners.

ISBN: 979-8-254-78222-3 (Paperback) eBook available on Kindle (ASIN assigned by Amazon KDP)

Printed and distributed via Amazon KDP Worldwide 2026

TABLE OF CONTENTS

PART I — FOUNDATIONS

Ch.01 — What Is a Project?
Ch.02 — Project Management Methodologies
Ch.03 — The Project Manager: Role, Competencies and Mindset

PART II — PROJECT INITIATION

Ch.04 — Project Charter and Initiation
Ch.05 — Stakeholder Management

PART III — PROJECT PLANNING

Ch.06 — Work Breakdown Structure
Ch.07 — Time Planning and Scheduling
Ch.08 — Budget Planning and Cost Management
Ch.09 — Resource Planning
Ch.10 — Risk Management
Ch.11 — Communication Management

PART IV — EXECUTION AND CONTROL

Ch.12 — Project Execution
Ch.13 — Change Management and Claim Management
Ch.14 — Quality Management
Ch.15 — Procurement Management

PART V — CLOSURE AND BEYOND

Ch.16 — Project Closure
Ch.17 — Project Management Office and Organizational Maturity
Ch.18 — Digital Project Management

Appendix A — Bibliography
Appendix B — Glossary of Terms
Appendix C — PM Templates Reference Index
Appendix D — PM Certifications Guide
About the Author

Introduction

Every great achievement begins as a project.

Why Project Management Matters More Than Ever

The bridge that spans a river. The software that connects millions. The factory that produces the machines of tomorrow. The hospital wing that will save lives not yet threatened. None of these emerge from good intentions alone — they are born from planning, from coordination, from the disciplined art of turning ideas into reality within the constraints of time, resources, and human energy.

This is the domain of Project Management.

In the decades since Frederick Winslow Taylor first articulated the idea of scientific management, and since the Apollo program demonstrated what organized human effort could achieve, Project Management has evolved from a niche engineering discipline into a universal professional language. Today, it is spoken in automotive factories and artificial intelligence laboratories, in hospital corridors and government ministries, in startups launching their first product and enterprises managing portfolios of hundreds of concurrent initiatives.

And yet — projects fail. They fail constantly, spectacularly, and at enormous cost.

The Standish Group's annual Chaos Report consistently reveals that more than half of all projects are delivered late, over budget, or with reduced scope. Research by McKinsey & Company shows that large IT projects run on average 45% over budget and 7% over time, while delivering 56% less value than predicted. In manufacturing and infrastructure, the numbers are equally sobering.

The question is not whether projects are difficult. They are. The question is: why do some project managers consistently deliver, while others consistently struggle?

This book is an attempt to answer that question.

What This Book Is

Project Management: From Initiation to Closure is a practitioner's handbook. It is not a theoretical treatise, nor a certification exam guide — though it will serve you well if you are pursuing PMP, PRINCE2, or PMI-PBA credentials. It is a working companion, designed to be read cover to cover by those new to the discipline, and consulted chapter by chapter by experienced professionals facing specific challenges.

The book draws on three sources of knowledge: established global frameworks — PMBOK (PMI), PRINCE2, ICB/IPMA, and Agile methodologies; decades of academic research on organizational behavior, decision theory, and complex systems; and direct practitioner experience accumulated across industrial automation, automotive engineering, cybernetics, and enterprise software implementation.

Every chapter contains not only concepts but tools — templates, matrices, checklists, and decision frameworks that can be applied immediately to real projects.

A Word on Methodology

There is a persistent debate in the project management community between the advocates of traditional, plan-driven approaches and the champions of Agile. This book refuses to take sides, because the debate is largely false.

Real projects exist on a spectrum. Some — a software product developed by a small team with evolving requirements — benefit enormously from iterative, adaptive approaches. Others — a construction project or a regulatory compliance program — require the discipline of detailed upfront planning. Most projects, in practice, are somewhere in between, and the modern PM must be fluent in both languages.

You will find both here. Not as competing philosophies, but as complementary tools in a single professional toolkit.

Who This Book Is For

This book is written for:

- Project managers at any stage of their career who want to deepen their practice
- Business analysts, team leads, and department heads who manage projects as part of a broader role
- Students of management, engineering, and information technology preparing for professional life
- Executives and sponsors who want to understand what good project management looks like — and how to support it
- Anyone who has sat in a project meeting and thought: there must be a better way to do this

How This Book Is Organized

The book is structured in five parts that mirror the natural lifecycle of a project:

Part I — Foundations establishes the conceptual bedrock: what a project is, what major methodologies exist, and what makes an effective project manager.

Part II — Initiation covers the critical early work: writing the project charter, defining objectives, and mapping your stakeholders.

Part III — Planning is the largest section, addressing the full spectrum of planning disciplines: scope (WBS), time, cost, resources, risk, and communication.

Part IV — Execution and Control deals with the reality of running a project: leading teams, managing change, maintaining quality, and controlling procurement.

Part V — Closure and Beyond covers the disciplines that many PMs neglect: formal closure, lessons learned, PMO setup, and the emerging digital tools reshaping the profession.

The appendices provide a comprehensive bibliography, a glossary, a templates index, and a guide to PM certifications.

A Note on the Author

The author has spent his professional life at the intersection of technology and organizational change — managing projects in automotive engineering, industrial automation, and AI systems development across Central Europe. The perspectives in this book are shaped by real projects, real failures, real recoveries, and the accumulated wisdom of teams who learned, often the hard way, what works.

This is not a perfect book. No book about a living discipline can be. Project management evolves as the world evolves. But the fundamentals — clarity of purpose, honest communication, disciplined planning, and the courage to adapt — remain constant.

Let us begin.

◆◆ How to Use This Book

- Read Parts I and II sequentially if you are new to project management.
- Jump to any chapter in Parts III–V when you need a specific tool or technique.
- Use the Templates Reference Index (Appendix C) to quickly locate a specific form or worksheet.
- The Glossary (Appendix B) covers both project management and technical terms from industrial automation, robotics, and cybernetics.

PART I

Foundations of Project Management

CHAPTER 1

What Is a Project?



Figure 1.1:The Triple Constraint — Scope, Time, and Cost with Quality at the center. Every project management decision operates within this framework.

> *Every great human achievement began as a project. The challenge is not recognizing that fact — it is understanding precisely what that means, and what it demands of the people responsible for turning ambition into reality.*

1.1 Defining a Project

The Project Management Institute defines a project as a temporary endeavor undertaken to create a unique product, service, or result. Three words in that definition carry the entire weight of the discipline: temporary, unique, and endeavor.

Project (PMI Definition)

A temporary endeavor undertaken to create a unique product, service, or result. A project has a defined beginning, a defined end, and produces something that has not existed before in exactly that form.

Temporary

Every project has a defined beginning and a defined end. When the objective is achieved — or when it becomes clear it cannot be achieved — the project closes. This is what distinguishes a project from an ongoing operation. A hospital's emergency department runs indefinitely; the construction project that built it had a start date, a completion date, and a handover moment.

Temporary does not mean short. A major infrastructure project may run for a decade. What matters is that there is a point at which the project team disbands, the budget is closed, and the deliverable passes into the hands of those who will operate it.

Unique

No two projects are identical. Even when an organization builds dozens of similar systems — the same product for different clients, the same software deployed in different countries — each project involves a different configuration of people, constraints, risks, and context. This uniqueness is what makes projects inherently uncertain and therefore demanding. It is also what makes project management a discipline in its own right, distinct from the management of repetitive processes.

Endeavor

An endeavor implies intentional, organized effort directed toward a goal. Projects do not happen by accident. They require deliberate planning, deliberate coordination, and deliberate decision-making under conditions of uncertainty. This intentionality is the raw material of project management.

1.2 Projects vs. Operations

Understanding the boundary between a project and an operation is foundational to understanding why project management exists as a distinct discipline.

Operations are ongoing, repetitive processes that sustain the organization's core business. They optimize for efficiency, consistency, and reliability. The goal of operational management is to run the same process, better and better, indefinitely.

Projects are temporary, unique endeavors that produce something new. They optimize for delivering a defined output within constraints of scope, time, and cost. The goal of project management is to bring something into existence that did not exist before, then

hand it over.

A manufacturing line that produces the same component every day is an operation. The initiative to design, procure, install, and commission that manufacturing line is a project. When the commissioning is complete and the line runs normally, the project closes and the operation begins.

This handover moment — from project to operation — is one of the most critical and frequently mismanaged transitions in organizational life. Projects that deliver technically but fail to prepare the receiving organization for operational reality have not truly succeeded.

Dimension	Project	Operation
Duration	Temporary, defined end	Ongoing, indefinite
Output	Unique product, service, or result	Repetitive, standardized output
Goal	Deliver something new	Sustain existing capability
Team	Assembled for the project, then disbanded	Permanent, stable workforce
Uncertainty	High — novelty creates unknowns	Low — process is known and refined
Success metric	On time, on budget, to specification	Efficiency, quality, throughput

1.3 The Triple Constraint: Scope, Time, and Cost

Every project operates within three fundamental constraints. Their relationship is the central structural fact of project management, and no practitioner can escape it.

Scope

Scope defines what the project will deliver. It is the totality of work required to produce the project's output — and nothing more. Scope has two dimensions: product scope (the features and functions of the deliverable) and project scope (the work required to create it). A clearly defined scope is the foundation on which all other planning rests. An undefined or poorly defined scope is the most reliable predictor of project failure.

Time

Time defines when the project will deliver. The schedule translates the scope into a sequence of activities with durations, dependencies, and deadlines. Time is often the most visible constraint — the one that executives watch most closely and stakeholders feel most directly. A project that delivers late may deliver perfectly on scope and within budget, yet still be regarded as a failure.

Cost

Cost defines the resources the project may consume. Budget is not simply a cap on spending — it is a statement about organizational priorities. Every budget decision is a decision about what matters. The PM who understands the cost constraint deeply — who knows where the money is coming from, what happens if it is exceeded, and what drives each major cost element — operates with a significant strategic advantage.

The interdependence

The three constraints are inseparable. Change one, and at least one other is affected. This is not a theoretical model — it is the governing reality of every project ever managed.

Scope increase — adding features or deliverables — requires more time, more cost, or both. If neither is available, quality suffers.

Time compression — an earlier deadline — requires either reduced scope or increased cost (adding resources). Sometimes both.

Cost reduction — cutting the budget — requires reduced scope, extended timeline, or acceptance of lower quality.

The PM's job is not to wish this interdependence away, but to make it visible to stakeholders so that decisions are made consciously. When an executive demands more scope, a shorter timeline, and the same budget simultaneously, the PM's obligation is to explain clearly what that combination would cost in quality — and to do so before the commitment is made, not after the failure occurs.

1.4 Quality as the Fourth Dimension

Modern project management adds Quality as a fourth dimension of constraint — one that sits at the center of the triple constraint triangle, affected by every change to scope, time, or cost.

Quality in a project context has a precise meaning: the degree to which the project's deliverables satisfy the requirements for which they were produced. This definition has two important implications.

First, quality is not the same as luxury or perfection. A deliverable that meets its specified requirements is a quality deliverable, even if those requirements are modest. Gold-plating — delivering more than was specified — is not a quality improvement. It is a scope violation that consumes resources without authorization.

Second, quality must be defined before it can be managed. Requirements, acceptance criteria, and quality standards must be established during planning, not discovered during closure. The Acceptance Protocol — the document that will formally confirm delivery — should be agreed upon before the project begins its execution phase.

1.5 Portfolio, Program, and Project

Organizations rarely manage projects in isolation. They operate within a hierarchy of complexity that determines how strategic decisions are made and how resources are allocated.

Project

The project is the unit of execution. It delivers a specific output within defined constraints. Project management asks: how do we deliver this specific thing, on time, on budget, and to specification?

Program

A program is a group of related projects managed in a coordinated way to obtain benefits not achievable by managing each project independently. Programs are about synergy and coordination. A program to modernize a company's logistics infrastructure might include projects for warehouse automation, fleet management software, and staff training — each a project in its own right, but interdependent in ways that require coordination above the project level.

Portfolio

A portfolio is a collection of projects, programs, and operations managed as a group to achieve strategic objectives. Portfolio management does not ask how to execute individual projects — it asks whether the organization is working on the right projects at all. It is the level at which resource allocation, strategic alignment, and investment decisions are made across the entire project landscape.

Level	Focus	Key question	Managed by
Portfolio	Strategic alignment	Are we doing the right things?	Executive leadership, PMO

Level	Focus	Key question	Managed by
Program	Benefit realization	Are we coordinating effectively?	Program Manager
Project	Delivery	Are we delivering correctly?	Project Manager

1.6 Project Lifecycle: Phases and Gates

All projects, regardless of methodology, pass through a recognizable sequence of phases. The specific labels vary across frameworks — PMBOK uses Initiating, Planning, Executing, Monitoring and Controlling, and Closing; PRINCE2 uses Starting Up, Initiation, and several delivery stages — but the underlying logic is consistent.

The five-phase model

1. Initiation: The project is formally authorized. The business case is validated, the project charter is issued, and the project manager is appointed. This phase ends when the organization commits resources to the project.

2. Planning: Detailed plans are developed for scope, schedule, budget, resources, risk, quality, and communication. Planning is not a bureaucratic exercise — it is the process of thinking through the project before executing it, while the cost of changing course is still low.

3. Execution: The work defined in the plan is performed. Teams are coordinated, deliverables are produced, and vendors are managed. Execution consumes the largest share of the project's resources.

4. Monitoring and Control: Performance is continuously measured against the plan. Variances are identified, analyzed, and addressed. This phase runs in parallel with execution throughout the project — it is not sequential to it.

5. Closure: The project is formally concluded. Deliverables are accepted and handed over, contracts are closed, resources are released, and lessons learned are documented.

Quality Gates

Between phases — and sometimes within them — organizations institute formal Quality Gates (also called Stage Gates or Phase Gates). A Quality Gate is a structured review point at which the project's readiness to proceed is assessed against predefined criteria.

At each gate, four outcomes are possible: Go (the project proceeds as planned), Waiver (the project proceeds with documented exceptions), Delay (the project is paused until

specific conditions are met), or Kill (the project is terminated because continuing is no longer justified).

The discipline of Quality Gates reflects a fundamental truth: the cost of terminating a failing project increases dramatically with every phase it is allowed to enter. A project that should be killed at the end of initiation — because the business case no longer holds, or because the risks are unacceptable — becomes enormously more expensive to kill after two years of execution. Gates exist to make the kill decision when it is still affordable.

Quality Gate

A formal review point between project phases at which the project's readiness to proceed is assessed against predefined criteria. Possible outcomes: Go, Waiver, Delay, or Kill (terminate). Quality Gates are a governance mechanism, not a bureaucratic obstacle — they protect the organization from continuing to invest in projects that no longer justify their cost.

Agile lifecycle

In Agile projects, the lifecycle is compressed into short, repeating iterations called sprints (in Scrum) or iterations (in SAFe and other frameworks). Each sprint contains its own miniature cycle of planning, execution, review, and retrospective. The strategic lifecycle — from initiation to closure — still exists; what changes is how the execution phase is structured internally.

The practical implication is that Agile projects generate usable deliverables early and often, allowing stakeholders to validate assumptions before the project has consumed its entire budget. This is a significant risk management advantage in environments where requirements are uncertain or likely to evolve.

1.7 Project Categorization

Not all projects are equal in complexity, risk, or strategic importance — and managing them as if they were is a common and costly mistake. Sophisticated organizations classify their projects before assigning management resources, allowing the level of oversight, governance, and PM experience to be calibrated to what each project actually demands.

Categorization dimensions

Projects are typically categorized along two dimensions:

- Complexity: How many stakeholders are involved? How many dependencies? How novel is the technology or approach? How significant are the organizational change implications?
- Size: Budget, duration, team size, and geographic scope.

The intersection of these two dimensions produces a classification matrix. A small, low-complexity project — a software patch, an internal process improvement — can be managed by a junior PM with a light governance framework. A large, high-complexity project — a multi-country system rollout, a major capital investment — requires a senior PM with deep experience, a formal governance structure, and rigorous risk management.

	Low complexity	High complexity
Small	Category A: Junior PM, light governance	Category B: Senior PM, standard governance
Large	Category C: Senior PM, standard governance	Category D: Expert PM, full governance

The exact category definitions vary by organization and industry. What matters is the principle: the PM assigned to a project should be chosen based on what that project demands, not on who happens to be available. Assigning a Category D project to a junior PM because the experienced ones are occupied elsewhere is not a resourcing decision — it is a risk decision, and it should be made explicitly.

1.8 The Project Lifecycle in an Industrial Context

In industrial and enterprise environments — manufacturing, logistics, engineering, IT systems integration — the project lifecycle is often framed as three distinct phases that span from first contact with the customer through the end of the product's operational life.

Sales phase

The Sales phase precedes the formal project. During this phase, the organization responds to customer interest — qualifying opportunities, preparing bids, negotiating contracts. A critical decision point in this phase is the Bid/No-Bid decision: the formal assessment of whether a particular opportunity is worth pursuing at all. Organizations that skip this discipline — pursuing every opportunity regardless of fit, margin, or deliverability — consume enormous resources on proposals they cannot win or should not win.

The Sales phase ends with contract signature. At this point, responsibility formally transfers from the sales organization to the project realization team. This Handover from Sales to Realization is a critical moment that many organizations handle poorly — institutional knowledge, customer commitments, and risk assumptions made during bidding must be explicitly transferred, not assumed.

Realization phase

The Realization phase is the project itself — from kick-off through delivery and acceptance. This phase encompasses initiation, planning, execution, and closure as described above. It ends when the customer formally accepts the deliverable and the project is closed.

Operation phase

After project closure, the deliverable enters its operational life. In many industries, the delivering organization remains involved — through maintenance contracts, service level agreements, or ongoing support. Understanding the Operation phase matters to the project manager for one specific reason: decisions made during the project have long-tail consequences in operation. A shortcut taken during installation, a requirement quietly dropped to meet a deadline, a training program delivered superficially — all of these create operational problems that persist for years after the project team has moved on.

 Concept Deep Dive — The Temporary Nature of Projects

The word temporary in the PMI definition carries more weight than it appears. It means the project has a defined authorization scope — a specific mandate to consume resources, make decisions, and produce outputs. When that mandate expires, so does the project team’s authority. Understanding this helps explain why projects that “run forever” are organizational pathologies, not project management successes: they are operations masquerading as projects, consuming project-level governance overhead indefinitely without the accountability mechanism that closure provides.

 Industry Insight — Project vs. Operation Across Sectors

Construction: Building a hospital is a project; running the hospital’s emergency department is an operation. The construction PM hands over to the facilities manager at completion — two fundamentally different management disciplines with different success metrics.

Manufacturing: Designing and commissioning a new production line is a project; running that line at target throughput is an operation. The project succeeds when the line is handed over; the operation succeeds when it runs reliably for years.

Software: Developing and launching a product is a project; operating, maintaining, and scaling it is a product operation. The most common failure mode is treating ongoing product development as a permanent project rather than as a product operation with its own governance model.

 Common Pitfalls — Misunderstanding Project Boundaries

• Scope confusion at initiation: Defining the project so broadly that it has no natural completion point — “improve the customer experience” is not a project; “reduce complaint resolution time from 5 days to 2 days by Q3” is.

• Project-operation blur: Failing to plan the handover to operations, so the project team is still supporting the deliverable 12 months after go-live because no one designed the operational model.

• Grade vs. quality confusion: Delivering a premium solution when a functional one was specified — gold-plating consumes budget without generating authorized value.

• Categorization neglect: Assigning a Category D project (large, high complexity) to a junior PM because the experienced ones are occupied — treating resourcing as a PM decision rather than a risk decision.

◆◆ Key Takeaways

• A project is a temporary, unique endeavor with a defined beginning and end, producing something that has not existed before in exactly that form.

• The Triple Constraint — Scope, Time, Cost — governs every project. Changing one constraint forces a change in at least one other.

• Quality is the fourth dimension, sitting at the center of the constraint triangle and affected by every change to the other three.

• Projects exist within a hierarchy: Portfolio (strategic alignment) — Program (benefit coordination) — Project (delivery execution).

• All projects pass through five phases: Initiation, Planning, Execution, Monitoring and Control, and Closure. Quality Gates between phases protect organizations from continuing to invest in failing projects.

• Project categorization — by complexity and size — ensures that PM experience and governance intensity are matched to what each project actually demands.

• In industrial contexts, the full project lifecycle spans Sales, Realization, and Operation — and the PM's decisions in Realization have consequences that extend into all three.

◆ Author’s Note — Practitioner Insight

In industrial automation and robotics projects, the distinction between project and operation is not merely academic — it is contractual. When our team commissions an automated guided vehicle system for a distribution center, the Preliminary Acceptance marks the moment we hand over a functioning system; the Final Acceptance, months later, marks the moment we confirm it has operated reliably under real production load. Between those two dates, we are simultaneously in project mode and operation mode. The lifecycle model in this chapter maps directly to that reality.

Project categorization matters enormously in this context. A firmware update to an existing AGV fleet is a Category A project; a full greenfield robotics deployment integrating 40 vehicles, a warehouse management system, and a new conveyor architecture is firmly Category D. Treating them the same way — as many organizations do — guarantees that one of them is over-managed and the other is catastrophically under-managed.

CHAPTER 2

Project Management Methodologies



Figure 2.2:The five major PM methodologies. From structured PMBOK and PRINCE2 to iterative Scrum and Kanban, and the Hybrid approach that dominates modern practice.

> *A methodology is a system of principles, practices, and procedures applied to a specific field. In project management, the right methodology — or the right combination — is not a philosophical preference. It is a practical decision with direct consequences for how the project is planned, executed, and governed.*

2.1 The PMBOK Framework

The Project Management Body of Knowledge (PMBOK®), published by the Project Management Institute (PMI), is the most widely adopted PM framework in the world. It defines project management through a structured body of knowledge that has evolved across seven editions since its first publication in 1996.

From processes to principles

The 6th edition organized project management into 49 processes across five process groups (Initiating, Planning, Executing, Monitoring and Controlling, Closing) and ten

knowledge areas (Integration, Scope, Schedule, Cost, Quality, Resource, Communication, Risk, Procurement, Stakeholder). This model provided an exhaustive, process-driven reference that became the basis for the globally recognized PMP certification.

The 7th edition (2021) made a significant conceptual shift: from prescriptive processes to foundational principles. It articulates twelve principles of project management — stewardship, team, stakeholders, value, systems thinking, leadership, tailoring, quality, complexity, risk, adaptability, and change — alongside eight performance domains. This shift acknowledges that no single set of processes fits every project, and that the profession has matured beyond the need for universal prescription.

PMBOK in practice

The PMBOK is best understood as a reference architecture, not a methodology to be followed mechanically. Experienced PMs use it selectively — drawing on the knowledge areas and processes most relevant to their project's context, and adapting or simplifying what is not. The PMP certification, earned by demonstrating mastery of PMBOK principles, remains the gold standard credential in project management globally.

PMBOK (Project Management Body of Knowledge)

A globally recognized standard and guide published by PMI. The 7th edition defines twelve principles and eight performance domains applicable across all project types and industries. It is descriptive — what good project management looks like — rather than prescriptive — exactly how to do it.

2.2 PRINCE2: Controlled Projects

PRINCE2 (Projects IN Controlled Environments) is a structured, process-driven methodology originally developed for UK government IT projects and now maintained by PeopleCert. It is characterized by seven principles, seven themes, and seven processes — and by an unusually clear separation between project governance (the Directing layer) and project management (the Managing and Delivering layers).

The seven principles

PRINCE2's principles are not optional decorations — they are the conditions that define whether a project is truly being managed under PRINCE2. The most important of them is Continued Business Justification: a project must have a valid business case throughout its entire lifecycle. If the business case is no longer valid — because costs have escalated, the market has changed, or the expected benefits can no longer be

realized — the project must be stopped. This ruthless pragmatism is PRINCE2's most valuable and most often ignored contribution to the profession.

The other six principles are: Learn from Experience, Defined Roles and Responsibilities, Manage by Stages, Manage by Exception, Focus on Products, and Tailor to Suit the Project Environment.

Strengths and limitations

PRINCE2 is particularly well-suited to large, complex projects in regulated environments — public sector, infrastructure, defense, financial services. Its explicit role definitions and governance structure make it an excellent framework for multi-organizational projects where accountability must be clearly assigned. Its weakness is that it can feel heavyweight for smaller projects, and its terminology can become a language barrier for teams unfamiliar with it.

PRINCE2 is not a replacement for PM skill. Its processes define what to do and when; they do not tell you how to lead a team, negotiate with a difficult stakeholder, or make a judgment call under uncertainty. The methodology provides the structure; the project manager provides the judgment.

2.3 Agile: Values, Principles, and Frameworks

The Agile movement did not emerge from a methodologist's office. It emerged from the collective frustration of software developers who had experienced, repeatedly, the consequences of building systems through lengthy, rigid, document-heavy processes that delivered the wrong thing perfectly.

The Agile Manifesto

In February 2001, seventeen software practitioners gathered in Snowbird, Utah, and produced a document of 68 words that changed an industry. The Agile Manifesto articulates four values:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

The manifesto's authors were careful to note: "while there is value in the items on the right, we value the items on the left more." Agile is not anti-documentation or anti-process. It is a set of priorities about where to invest attention when trade-offs must be made.

Scrum

Scrum is the most widely adopted Agile framework. Developed by Ken Schwaber and Jeff Sutherland in the early 1990s and codified in the Scrum Guide (updated most recently in 2020), it organizes work into time-boxed iterations called sprints, typically two weeks in length.

Scrum defines three roles, three artifacts, and five events:

- Roles: Product Owner (accountable for product value and backlog), Scrum Master (accountable for Scrum process adherence and impediment removal), Developers (accountable for creating the increment)
- Artifacts: Product Backlog (the ordered list of all work to be done), Sprint Backlog (the subset of backlog selected for the current sprint), Increment (the sum of all completed backlog items, meeting the Definition of Done)
- Events: Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective, and the Sprint itself

Scrum's power lies in its cadence. By delivering a potentially shippable increment every two weeks, it forces regular stakeholder engagement, early feedback, and continuous course correction. Problems that would surface in month eight of a waterfall project surface in week two of a Scrum project — when they are still cheap to fix.

Kanban

Kanban is a visual workflow management method originating in Toyota's production system, adapted for knowledge work by David Anderson in the 2000s. Unlike Scrum, Kanban prescribes no fixed iterations, no roles, and no ceremonies. Its core mechanism is the Kanban board — columns representing workflow states (To Do, In Progress, Review, Done) — combined with a critical constraint: Work In Progress (WIP) limits.

WIP limits prevent the accumulation of partially completed work that clogs a team's capacity and obscures real progress. A team that limits its In Progress column to three items cannot start new work until existing work is finished. This sounds simple; its consequences are profound. Teams that enforce WIP limits consistently deliver faster and with fewer defects than teams that allow unlimited work to accumulate.

SAFe and enterprise Agile

For large organizations attempting to apply Agile principles across multiple teams and programs, frameworks like the Scaled Agile Framework (SAFe) and Large-Scale Scrum (LeSS) provide coordination structures above the individual team level. SAFe

introduces the Agile Release Train — a virtual team of 50 to 125 people aligned to a common mission and cadence — as the primary coordination mechanism for enterprise-scale delivery.

Enterprise Agile scaling is genuinely complex, and implementation failures are common. The most frequent failure mode is adopting Agile ceremonies and terminology without changing the underlying management behaviors — a condition sometimes called "Agile in name only" or "Agile in Name Only" (waterfall with Agile labels).

2.4 Waterfall: Sequential Delivery

The term Waterfall was never used by its originator — Winston Royce, who described it in a 1970 paper as a model he considered fundamentally flawed — but it has become the shorthand for any sequential, phase-based project approach in which each phase must be substantially complete before the next begins.

In a Waterfall project, requirements are defined in full before design begins, design is complete before development begins, and development is complete before testing begins. This approach offers predictability and auditability. When the requirements are genuinely stable — as in construction, manufacturing, or regulatory compliance — it is appropriate and effective. When requirements are uncertain or likely to evolve, it is a reliable path to delivering the wrong thing on time and on budget.

2.5 Hybrid Approaches

Most real-world projects do not sit cleanly at either end of the traditional-Agile spectrum. Hybrid approaches — sometimes called "Agile in Name Only" pejoratively, but more accurately described as tailored or adaptive methodologies — combine elements of sequential and iterative delivery in ways calibrated to the specific project context.

Common hybrid patterns

A typical hybrid pattern uses a traditional approach for the project's structural phases (initiation, planning, procurement, governance) and an Agile approach for execution (iterative delivery of software or other configurable components). This preserves the governance benefits of sequential planning — a defined scope, an approved budget, clear accountability — while gaining the flexibility and feedback benefits of iterative execution.

Another common pattern applies Agile for internal development work while maintaining a Waterfall interface to the customer — providing the structured

milestones and formal deliverables the customer expects, while using Scrum sprints internally to produce them.

Why hybrid dominates

PMI's annual Pulse of the Profession surveys consistently show that the majority of high-performing organizations use hybrid approaches. This is not indecision — it is sophistication. The project manager who is fluent in both traditional and Agile approaches, and who can identify the right blend for each project's context, is significantly more effective than one committed to either extreme.

Criterion	Traditional (Waterfall)	Agile	Hybrid
Requirements	Stable, well-defined	Evolving, unclear	Mixed
Client involvement	Defined touchpoints	Continuous	Structured + iterative
Risk tolerance	Low (regulated)	Higher	Medium
Team size	Any	Small to medium	Any
Delivery rhythm	Single final delivery	Frequent increments	Milestones + increments
Typical domains	Construction, manufacturing, compliance	Software, product development	IT, engineering, transformation

2.6 IPMA and the Competence-Based Model

The International Project Management Association (IPMA) approaches project management from a fundamentally different angle than PMI or PRINCE2. Where those frameworks define what to do, IPMA defines what a competent PM should know, be able to do, and be like. Its certification model is based on demonstrated competence, not examination performance.

The Individual Competence Baseline (ICB)

The ICB v4.0 organizes competence across three domains:

- People competences: Self-reflection, leadership, teamwork, conflict resolution, resourcefulness, negotiation, results orientation, and others. These are the behavioral competences that determine how a PM works with people.
- Practice competences: The technical PM skills — scope, schedule, quality, finance, resources, procurement, planning, risk management, and stakeholder

engagement.

• Perspective competences: The contextual competences that connect the project to the organization and its environment — strategy, governance, compliance, power and interest, and culture.

IPMA certification levels

IPMA certifies at four levels, from project team member (Level D) through Senior Project Manager (Level C), Project Manager (Level B), to Director of Projects (Level A). The upper levels require a portfolio interview and assessment by certified assessors — making IPMA Level A one of the most rigorous PM credentials available.

IPMA certification is particularly prevalent in Germany, Austria, Switzerland, the Czech Republic, and other continental European countries, where its competence-based model aligns well with engineering culture and professional development traditions.

2.7 Choosing the Right Methodology

The choice of methodology is not an academic exercise. It has concrete consequences for how the project is governed, how the team is organized, how stakeholders are engaged, and ultimately whether the project succeeds.

Factors that drive the choice

• Requirement stability: The more stable and complete the requirements, the more suitable a sequential approach. The more uncertain or likely to evolve, the stronger the case for iteration.

• Regulatory environment: Regulated industries — finance, healthcare, aerospace, nuclear — often require documented, auditable processes that favor traditional approaches.

• Customer expectations: Some customers want a defined deliverable at a defined date; others want continuous collaboration and regular increments. Both are legitimate. The methodology must match the customer's engagement model.

• Team experience: A team new to Scrum attempting to implement it on a high-stakes project without coaching is a significant risk. Methodology choice must account for the team's actual capability, not its aspired capability.

• Organizational culture: A hierarchical, plan-driven organization implementing Agile without cultural change is attempting a methodology transplant without the supporting immune system. Methodology and culture must align.

The meta-skill

The most valuable PM skill with respect to methodology is not mastery of any single framework — it is the ability to select, adapt, and combine approaches based on the specific demands of the project at hand. This requires genuine familiarity with multiple frameworks, the intellectual honesty to recognize which one fits, and the organizational influence to implement it effectively.

Methodologies are tools, not identities. A PM who insists on Agile for a regulated construction project, or Waterfall for a rapidly evolving software product, is letting preference override judgment. The project's needs — not the PM's comfort zone — must drive the methodology decision.

◆◆ Key Takeaways

- PMBOK provides a globally recognized reference architecture; its 7th edition shifted from prescriptive processes to foundational principles applicable across all project types.
- PRINCE2 is distinguished by its emphasis on continuous business justification — a project without a valid business case must be stopped, regardless of how far it has progressed.
- Scrum organizes delivery into two-week sprints with defined roles, artifacts, and events, enabling rapid feedback and course correction. Kanban manages flow through WIP limits without fixed iterations.
- Hybrid approaches — combining sequential governance with iterative execution — are the current standard in high-performing organizations. Fluency in both traditional and Agile approaches is a core PM competence.
- IPMA's competence-based model assesses what PMs know, can do, and are — across people, practice, and perspective domains — rather than testing process knowledge alone.
- Methodology choice is driven by requirement stability, regulatory environment, customer expectations, team experience, and organizational culture. No single methodology is universally optimal.

◆ Author’s Note — Practitioner Insight

Hybrid methodology is not a compromise in robotics and automation projects — it is the only approach that works. The hardware commissioning timeline is fixed by physics and safety certification requirements: you cannot sprint your way through factory acceptance testing of a safety-critical autonomous system. But the software that controls that hardware — the navigation algorithms, the fleet management logic, the interface to the warehouse management system — benefits enormously from iterative development and continuous stakeholder feedback.

Publishing Team

Project Management: From Initiation to Closure

Author, publisher and co-authors

Author/Publisher: **Radim Kaufmann**

Editor-in-Chief: **Radim Kaufmann**

Co-Authors: **Claire Whitaker, Spencer Reed, Julian Hartwell, Tessa Monroe, Gavin Rhodes, Naomi Fletcher, Wesley Porter**

Editorial team

Managing Editor: **Erin Caldwell**

Acquisitions Editor: **Camille Foster**

Series Editor: **Audrey Nolan**

Project Editor: **Owen Barrett**

Section Editor: **Marissa Cole**

Developmental Editor: **Nora Callahan**

Line Editor: **Lydia Shaw**

Copy Editor: **Maya Chen**

Proofreader: **Leah Winters**

Research and specialist preparation

Editorial Researcher: **Priya Bennett**

Fact-Checker: **Isaac Merritt**

Indexer: **Cora Delaney**

Instructional Designer: **Keisha Benton**

Subject Matter Reviewer: **Elena Sandoval**

Assessment Specialist: **Aisha Coleman**

Technical Editor: **Andre Wallace**

Art and design

Art Director: **Vera Hollander**

Book Designer: **Jasper Vaughn**

Book Cover Designer: **Roman Blake**

Photo Editor: **Celeste Rowan**

Illustrator: **Hazel Corbin**

Typesetting and production

Typesetter: **Fiona Ashby**

Production Manager: **Meredith Sutton**

Ebook Developer: **Devin Cho**

Rights and distribution

Rights and Permissions Manager: **Tara Mendoza**

Kaufmann Creative AI Studio©

Creative Director: **Nolan Pierce**

Concept Artist: **Ivy Carver**

Educational Illustrator: **Lucia Vega**

Digital Imaging Specialist: **Felix Archer**

Original illustrations and visual concept © Kaufmann Creative AI Studio. The original authorship information and licenses for third-party photographs, reproductions and other works remain stated with the respective works and in this book’s image credits.

Publisher team

Expanded credits · November 17, 2026