

Turning Projects into Products: Making Delivery Stick in Complex Organizations ¹

Why project and program leaders should treat go-live as the start of adoption, not the end of delivery.

Sudhakar Shivaraju

In large organizations, many initiatives begin with strong sponsorship, a credible business case, and a committed project team. Yet a year after go-live, the reality can look very different. The new process is only partly adopted, reports still need manual reconciliation, and teams quietly keep pieces of the old way of working because it feels safer or faster. This is where many projects fall short, not because the project team failed to execute, but because the organization treated the change as something to launch rather than something to operate, measure, and improve.

A product mindset can help close that gap. This does not mean turning every internal initiative into a software product or borrowing start-up language that does not fit the environment. It means applying disciplines that matter in any serious delivery setting: clarity of outcome, visible ownership, user feedback, adoption measures, and continued improvement after the formal milestone has passed. In project and program environments, those disciplines can be the difference between a completed initiative and a durable capability.

A useful starting point is to define the outcome in operational terms. Many project charters still describe success as a deliverable: implement a system, roll out a process, migrate data, or complete training. Those deliverables matter, but they are not the same as business change. The more useful question is simple: what should be different in the day-to-day work six or twelve months after go-live? If the answer is fewer manual reconciliations, faster onboarding of new projects, or more reliable portfolio reporting, then those outcomes need

¹ How to cite this article: Shivaraju, S. (2026). Turning Projects into Products: Making Delivery Stick in Complex Organizations, commentary, *PM World Journal*, Vol. XV, Issue VIII, August.

to be named early and revisited often. Once the outcome is explicit, decisions about scope, sequencing, and trade-offs become easier to defend.

For example, if a new PMIS is intended to improve portfolio reporting, success should not be measured only by whether the system went live on time. It should also be measured by whether project teams enter data consistently, whether leaders trust the reports enough to use them in decision reviews, and whether manual spreadsheet reconciliation begins to decline. Those signals say more about the value of the change than the launch milestone alone.

Ownership is the next practical issue. In complex organizations, accountability can become diluted across steering committees, PMOs, technology teams, operations leaders, and control functions. Each group may be doing its part, but no one is clearly responsible for whether the new way of working takes hold. A product-style model does not remove governance. It makes accountability more visible. One person should be responsible for the end-to-end outcome, supported by a cross-functional team that can make practical decisions, resolve competing priorities, and keep the focus on value after implementation.

Governance also has to support learning, not just reporting. Traditional project governance is often built around schedule, budget, risks, issues, and status indicators. These are necessary, especially in regulated or capital-intensive environments, but they do not always show whether the change is working in practice. Product-oriented governance still tracks delivery health, but it also brings user behavior and adoption into the discussion. The questions are practical:

- Are people using what we built in the way we expected?
- Where are they creating workarounds?
- What feedback are we hearing from front-line users?

In practice, these answers may come from usage data, support trends, exception reports, surveys, or direct conversations with front-line teams. The source matters less than the habit of looking. If adoption feedback appears only in a lessons-learned session after the project is closed, it is too late to influence the outcome. The better approach is to make these signals part of the regular governance rhythm while the team still has the ability to respond.

Adoption also needs to be treated as real work, not as a communication task added near the end. Many project teams have seen the same pattern: a go-live date is approaching,

technical delivery is consuming attention, and change management becomes a training deck, a few announcements, and a support mailbox. That may be enough to inform people, but it is rarely enough to change behavior. A product mindset treats adoption as part of the product itself. Communication, training, role impacts, support processes, and success measures should be designed early, owned clearly, and adjusted as the organization learns what users are actually doing.

The same mindset changes how leaders view go-live. In many organizations, go-live is treated as the finish line because that is how the project plan, budget, and governance calendar are structured. But from the user's point of view, go-live is when the real test begins. The process meets operational pressure. Data quality issues surface. Edge cases appear. Teams discover where the design fits their work and where it does not. A small, stable team after rollout can make a significant difference by handling feedback, refining features, addressing recurring issues, and preventing the organization from drifting back to old habits.

This does not require dismantling existing project, PMO, or portfolio structures. Those disciplines remain important. The shift is one of emphasis. Instead of managing only toward delivery against an approved plan, leaders also manage toward adoption and sustained results. A practical starting point is to select a small number of initiatives where the organization defines behavioral outcomes, assigns a product-style owner, and adds adoption and user feedback to the governance cadence. The goal is not to create a new layer of terminology. It is to create a more honest operating model for change.

It is also important to be realistic about constraints. Large enterprises and regulated industries have valid requirements around approvals, controls, auditability, data quality, and documentation. Product thinking should not be used as an excuse to bypass those obligations. The more useful approach is to bring risk and control partners into the lifecycle earlier, when the problem and solution are still being shaped. That reduces the familiar late-stage tension between moving quickly and meeting governance expectations. In mature environments, speed and control are not opposites. Both improve when the work is better understood earlier.

The longer-term benefit is not simply better project execution. It is a clearer way to judge whether change is producing the intended result. When initiatives are connected to measurable outcomes, when one owner is accountable for the end-to-end experience, and when governance includes adoption and feedback, organizations can make better decisions

about what to scale, what to adjust, and what to stop. Completion metrics still matter, but they are no longer enough on their own.

For organizations running many large initiatives at the same time, this is not a small distinction. When adoption is unfinished, the next project often starts by fixing the gaps left by the last one. That is expensive, frustrating, and avoidable. This way of working gives project and program leaders a practical way to stay with the change after launch, see where it is working, and make adjustments before old habits return. In the end, the point is not to make project teams speak like product teams. The point is to make delivery last.

About the Author



Sudhakar Shivaraju

Atlanta, GA, USA



Sudhakar Shivaraju is a project and program management leader with experience spanning manufacturing, capital programs, and enterprise technology. He currently leads governance and systems strategy for a global real estate technology platform at a major financial institution. He holds a PMP certification and a Master of Science in Mechanical Engineering, and writes on governance, data quality, and AI readiness in enterprise project management and real estate technology environments.

The views expressed in this article are those of the author alone and do not represent the views of the author's employer.