



REAL AUTONOMY NEEDS EMBEDDED GOVERNANCE

Why MAHD Has Moved Beyond “Empowered Teams” as an Agile Principle

BY DORIAN SIMPSON

EXECUTIVE SUMMARY

Empowered teams were an important step forward. But in hardware development, many of the most significant decisions extend beyond a single team. Embedded Governance provides a more practical, effective approach-giving teams the freedom to act while ensuring the right people, information, and tradeoffs are in place for complex, cross-functional decisions. The result is faster, higher-quality decisions, reduced risk, and stronger product and business outcomes.

The Problem with Empowerment

Few ideas are more closely associated with agile than autonomous, self-organizing teams. The premise is compelling: move decisions closer to the people doing the work, remove unnecessary barriers, and trust capable teams to determine the best path forward.

That was an important correction to development environments in which decisions moved slowly through functional silos, layers of approval, and infrequent management reviews. Agile rightly challenged the assumption that management must direct every significant choice.

MAHD has always embraced that intent. But hardware development has also made one limitation increasingly clear: many consequential decisions do not belong entirely to the team—or entirely to management.

A technical choice may simultaneously affect customer value, product economics, architecture, manufacturing, supply chain, regulatory obligations, launch timing, capital investment, and future platforms. The people closest to the work may understand the technical issue best, while others hold essential context about the broader consequences.

In those situations, asking only how to empower the team is too narrow. The more useful question is:

How do we ensure that the right people make the right decisions, with the right information, at the right time?

That is the idea behind Embedded Governance. It is not a retreat from agility. It is an attempt to make agility more realistic and effective in complex product development.



THE EVOLUTION OF DECISION-MAKING: FROM CENTRALIZED CONTROL TO CONTEXTUAL FREEDOM.

A Principle Must Change Something

Organizations often adopt ideas such as empower teams, embrace change, collaborate closely, focus on the customer, and learn continuously. Few people disagree with those statements. The harder test is whether a principle helps when priorities conflict, uncertainty is high, or the right path is unclear.

“Empower the team” is useful when the problem is unnecessary approval or authority held too far from the work. But it provides less guidance when a decision crosses organizational boundaries. Empowered to decide what? Product scope? Platform architecture? Regulatory risk? Major investment? Launch timing? Resources shared with another program?

Principles matter only when they change behavior.

If a principle does not help people navigate those situations, it risks becoming aspirational language rather than an operating guide.

If a principle does not help people navigate those situations, it risks becoming aspirational language rather than an operating guide.

MAHD’s move from Empowered Teams to Embedded Governance reflects that standard. The intent is not to introduce new terminology for its own sake. It is to describe more accurately how strong teams and leaders make difficult decisions together.

Why Hardware Changes the Equation

Autonomy is easiest when the work and the consequences of decisions can remain largely within a team.

Software can sometimes support this model well. Teams may be able to design, build, test, deploy, observe, and adjust a capability quickly. Changes may be introduced incrementally and, in many cases, reversed at relatively low cost.

Hardware decisions are often harder to contain.

A component, supplier, material, architecture, interface, prototype, or manufacturing-process choice can affect tooling, unit cost, qualification, quality, regulatory approval, service requirements, launch timing, other engineering teams, or future products. The cost of changing direction can rise sharply once suppliers are committed, tooling is released, certifications are underway, or production begins.

Consider a team comparing two technical approaches. One offers significantly higher performance but requires a new supplier, additional tooling, greater technical risk, and months of qualification work. The other provides less performance but protects the launch schedule, preserves the manufacturing system, and requires less investment.

Engineering may be best positioned to judge technical feasibility. Product management may understand whether customers will value the added performance. Operations may see manufacturing and supply implications. Finance may understand margin impact. Leadership may need to weigh investment and timing against other portfolio priorities.

The goal is not maximum autonomy. It is better decision-making.

The choice is not merely technical. It is simultaneously a product, business, investment, timing, and portfolio decision.

Leadership involvement should not automatically be treated as interference. Executives are accountable for outcomes that often extend beyond the project team's line of sight. A seemingly small design choice can materially affect margin, revenue, market timing, supply exposure, regulatory risk, or future platform flexibility.

Of course, leadership engagement can become disruptive when it appears as unstructured intervention, changing preferences, or late-stage direction. The answer is not to keep leaders away from the work. It is to make their involvement more purposeful, timely, and evidence-based.

Why Guardrails and Phase Gates Are Not Enough

Many organizations have tried to balance control and autonomy through guardrails or "bounding boxes." Leadership defines acceptable ranges for cost, timing, performance, investment, scope, or risk; teams are then free to operate within them.

This is useful. Clear boundaries can prevent unnecessary approval cycles and give teams confidence to act.

But boundaries are not governance.

First, the variables interact. A project can technically remain within the acceptable range for cost, schedule, performance, risk, and margin while landing at the least desirable end of all five. Each parameter may be "inside the box," yet the combined outcome violates the strategic intent.

Second, boundaries age. A range established early in development may become less relevant as customer insight, technical evidence, competitor actions, market conditions, or business priorities change.

Third, a limit without its rationale can constrain judgment rather than improve it. A team may know the target cost but not why it matters, which tradeoffs are acceptable, or when additional cost would be justified by greater customer value or faster market entry.

Traditional phase-gate governance solves a different part of the problem, but often introduces its own friction. Leaders do need visibility. Major investments deserve review. Risks and strategic alignment matter. The problem is when governance is concentrated into large, infrequent events.

Teams prepare for the gate instead of the decision. Complex realities are compressed into presentations. Difficult questions wait for scheduled reviews. Leaders are asked to make large judgments from periodic snapshots. Risks can become less visible until a solution is ready, while important redirection arrives late.

This creates a false choice: centralize decisions and slow the work, or push authority outward and expect teams to resolve complex organizational tradeoffs independently.

Embedded Governance offers a third path.

Embedded Governance: A Better Operating Model

Embedded Governance treats governance as a shared decision system built into the normal flow of development—not as a gate, approval layer, or occasional management event.

It starts with strategic intent. Teams need more than requirements and deliverables. They need to understand the outcomes that matter, the customers being served, the advantage the product is expected to create, the constraints that are real, and the tradeoffs the organization is willing to make.

It also requires clearer decision paths. Not every choice deserves the same level of participation. Routine, reversible decisions with limited external impact should generally stay close to the work. Decisions involving major investment, system-wide consequences, enterprise risk, market timing, or strategic commitments require broader perspective.

MAHD describes autonomy as contextual freedom.

Autonomy should reflect the circumstances and consequences of the decision. When that context is clear, more decisions can remain with the team because the basis for making them is already understood.

A low-cost, reversible prototype choice with little effect outside the team may require no leadership involvement. A platform architecture decision affecting multiple products, long-term investment, regulatory strategy, or launch timing deserves a different conversation.

The distinction is not based on hierarchy alone. It reflects uncertainty, reversibility, investment, system impact, strategic importance, and the cost of being wrong.

This is not inconsistent empowerment. It is intelligent governance.

What Embedded Governance Enables

When done well, Embedded Governance should feel lighter, not heavier. Its value is not more process; it is better coordination around the decisions that matter.

Greater team freedom. When strategic intent, boundaries, and decision rights are understood, teams can act decisively without repeatedly seeking permission.

Faster, better decisions. Clear paths reduce time spent debating who should decide, escalating too late, or reopening choices that lacked the necessary context.

More useful leadership engagement. Leaders can focus on emerging tradeoffs, organizational constraints, resource needs, and consequential decisions rather than inserting themselves into routine technical work.

Earlier visibility of risk and tradeoffs. Teams are encouraged to surface uncertainty and cross-functional implications before they become crises or expensive commitments.

Greater adaptability. Governance becomes part of how the organization learns. New evidence can change the decision, the boundary, or even the direction without waiting for the next formal gate.

This creates an important paradox: clearer governance can create greater autonomy. When teams know where they have freedom, they can move faster. When leaders know where their involvement adds value, they have less reason to intervene everywhere.

Making Embedded Governance Actionable

Within MAHD, it is reinforced through the broader operating model rather than through a new layer of administration. Strategic alignment, transparent tradeoffs, recurring integration, rapid learning, and visible decision paths all contribute to the governance fabric.

A Vision Brief establishes shared intent. Customer Outcomes keep attention on value rather than only deliverables. The Focus Matrix makes priorities and tradeoffs explicit. Big Rocks focus attention on the uncertainties most likely to change the path forward. Divergent-convergent cycles encourage exploration before commitment. Context-Based Good Enough helps teams decide when enough evidence exists to move forward. IPAC planning and reviews create recurring opportunities for integration, prototyping, customer insight, alignment, and meaningful decisions.

Management check-ins can then focus less on activity reporting and more on what has been learned, which assumptions remain uncertain, what decisions are approaching, where tradeoffs are emerging, and what support is needed.

Embedded Governance only matters if it changes how work happens.

These practices are not governance added on top of development. Together, they embed governance into the way teams and leaders work.

Moving Beyond Empowerment

Agile was right to challenge excessive control. Decisions should be made as close to the work as practical, and capable teams should have meaningful ownership, freedom, and trust.

But autonomy is not a universal operating model, particularly in complex hardware development. Nor does “empower the team” fully explain how to handle decisions whose consequences extend across customers, systems, functions, investments, and portfolios.

Embedded Governance does not mean more meetings, approvals, or management intervention. It replaces disruptive governance with continuous alignment, unclear empowerment with contextual freedom, delayed escalation with visible decision paths, and episodic gates with timely transparency.

Teams retain ownership and leaders retain accountability, contributing where their knowledge, authority, and perspective are most useful. Moving beyond empowerment is not about giving teams less freedom. It is about creating the conditions for more effective freedom—and better decisions, at the right level, at the right time, with the right people involved.

About the MAHD Framework

The MAHD Framework is a product-development operating model designed for hardware-based products and other complex development environments where physical integration, long lead times, specialized resources, cross-functional dependencies, and persistent uncertainty can make conventional project methods — and software-centric agile practices on their own — difficult to apply directly.

The framework brings strategy, Customer Outcomes, cross-functional decision-making, flexible fidelity prototyping, risk and dependency management, vertical-slice learning, and structured adaptation into one system. Developed through more than a decade of field application, MAHD provides shared Guiding Concepts, Decision Guides, training, tools, and coaching intended to help organizations learn faster, implement more effectively, and sustain the operating behaviors described in this paper. Learn more at www.MAHDFramework.com.

AI Use & Editorial Transparency

Generative AI was used to support editorial review, research synthesis, wording refinement, and selected graphics. All content, claims, interpretations, terminology, and visual direction were reviewed, edited and approved by the author. All MAHD models, concepts, terminology and field data are the work and experience of MAHD Framework LLC.