Whitepaper

Ensuring clinical trial agent performance with continuous governance

Medable

www.medable.com

According to seven different market research organizations, the “Agentic AI in clinical trials market” is expected to grow at a compound annual growth rate (CAGR) of anywhere between 12.5% to 43%. While the estimates obviously vary depending on who you ask, one thing remains crystal clear, agentic AI in clinical trials isn’t slowing down.

As sponsors and CROs increasingly implement artificial intelligence, many organizations approaching clinical AI face real anxiety about the clinical trustworthiness of AI.

Specifically, they ask whether it will stay compliant? Will there be a traceable record for our auditor? How will I know why a decision was made, and by whom? These are not questions about technology. They are questions about governance.

Trust in clinical AI cannot be established at deployment and assumed thereafter. It must be maintained continuously, because AI is adaptive by design. Model providers push updates. Knowledge bases evolve. Workflows shift. Outputs can change over the course of a trial, subtly and silently, if no one is looking.

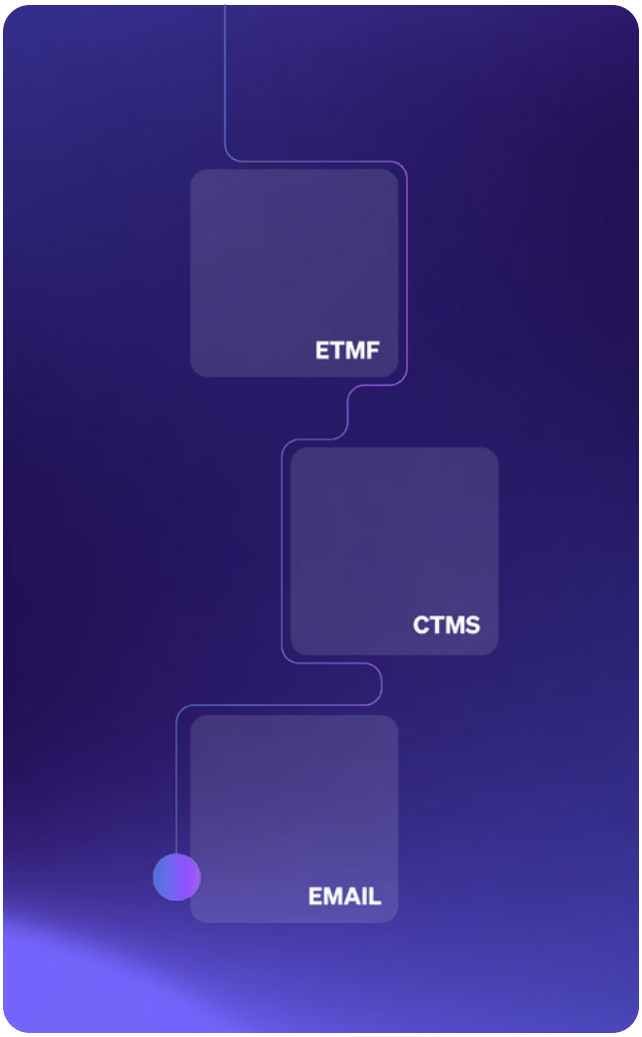
Here at Medable, our agentic AI is designed to continuously evaluate and adjust its own performance. This is because staying clinically capable isn’t a milestone to meet, it must remain ongoing.

Why trust is the real question in clinical AI

A different kind of software
Clinical AI agents are not a faster version of traditional validated software. The difference is structural, and impacts how AI performance validation works.

Traditional validated systems are deterministic. The same input always produces the same output. That property is what makes one-time validation sufficient: once you have demonstrated that behavior is correct, and the system is locked, behavior stays correct. The validation event is the governance. Additionally, data outages, software updates and business scheme logic updates in traditionally validated systems can still introduce variation and error, making them not as predictable as often thought.

Clinical AI agents are adaptive by design. As models improve, connected data sources change, and workflows evolve, so do their outputs. That adaptability is precisely what makes AI valuable over the life of a clinical trial. However, the need for ongoing validation is not unique to AI. Even traditional deterministic software undergoes UAT, regression testing, and validation following software releases, configuration changes, or integration updates. AI extends this principle. Rather than validating only at discrete milestones, trust must be established through continuous monitoring, validation, and oversight as the system evolves.



Traditional validated software	Clinical AI agents
Same input, same output. Always.	Outputs shift as models, data, and workflows evolve. Simply put, the answer will be wrapped in a different way each time.
One-time validation is sufficient to get started. However, it still needs verification for each and every update or data outage.	Continuous validation is required.
Behavior is fixed at release, and then for each subsequent update, as opposed to AI, which can change logic in between release to adapt.	Behavior is adaptive by design.
Governance is perceived as a “one-time-only” event for each release or update.	Governance is an ongoing operation.

The questions clinical teams are actually asking

Pharma and biotech organizations are approaching agentic AI with genuine enthusiasm and equally genuine concern. The concerns tend to cluster around the same four questions:

- **Will my compliance team be able to sign off on this?**
- **What happens when the model changes and no one notices?**
- **How do I show an auditor what the agent was doing in month nine?**
- **If something goes wrong, will I be able to trace it?**

These types of questions are governance questions, and in many non-clinical AI platforms, governance is either missing or bolted on after deployment as a configuration layer. That approach fails in regulated clinical environments, where evidence of control must be continuous.

What trustworthy clinical AI actually requires

Four properties together constitute genuine trustworthiness for a clinical AI deployment. Medable delivers all four as a unified, native platform:

1

Version control.
Each and every change made to the AI generates a new version, and a historical version of exactly how it performed previously, ensuring rollbacks and changes are capably logged.

2

Continuous performance validation.
Not a snapshot at deployment, but ongoing, scored evidence that the agent is performing as expected across every dimension relevant to its clinical function.

3

Complete, immutable traceability.
Every change to every component of the agent, with a full audit trail that can never be overwritten and can be exported at any time.

4

A governed lifecycle from the start.
Governance built into the platform architecture from day one, not layered on top of an existing system and not dependent on a third-party vendor.

The rest of this paper explains how each of these properties is delivered, why the alternative approaches fall short, and what continuous governance looks like in practice for clinical teams.

Understanding how clinical AI actually behaves

Adaptive AI is not unstable AI
There is a tendency in early conversations about clinical AI governance to frame adaptability as a risk category. Simply put, some worry that the AI might change, and that this is a problem to be managed. That framing is not accurate, and it leads to the wrong governance posture.

Adaptability is a design characteristic. It is what allows a clinical AI agent to remain useful as a trial evolves, as protocols change, and as the underlying model improves over time. The right governance posture does not resist adaptation. Instead, it makes adaptation visible, measurable, and correctable. That is precisely what Medable's agentic clinical trial platform is designed to do.

The more accurate framing is that clinical AI agents experience operational behavior shifts over time, and those shifts are natural. Some are driven by external changes the sponsor did not initiate. Some are the result of deliberate improvements by the clinical team. All of them need to be detected, assessed, and documented. The platform that can do that continuously is the platform you can trust.

What causes operational behavior to shift

Any deployed clinical AI agent will encounter four sources of change over the course of its deployment:

Model updates

Upstream model providers push changes that can alter response patterns without any action by the sponsor. These are often improvements, but they can affect how an agent behaves in ways that are not immediately visible.

Knowledge base changes

Study protocols, standard operating procedures, regulatory guidance, and connected data sources evolve over the course of a trial. An agent grounded in that knowledge will shift as the knowledge shifts.

Workflow and connector changes

Tools, triggers, and integrations change as the clinical operation matures. An agent whose tool-use behavior was validated against an earlier workflow configuration may behave differently after an update.

Prompt and configuration changes

Intentional improvements made by the team are a normal part of operating a clinical AI agent. They still require controlled, documented change management in a regulated environment.

None of these sources of change are problems to be prevented. They are realities to be governed. The platform that detects them, scores them, alerts the team, and documents the response is the platform that makes adaptation safe.

What undetected operational behavior shift looks like

An ICF review agent is deployed and scores 94% accuracy across the response accuracy dimension at baseline. In week six, a model provider pushes an update. The agent's accuracy on informed consent document review queries drops to 71%. No one has made a change. No alert has been triggered. No evaluation has run.

Without continuous monitoring, that degradation is invisible until a site coordinator surfaces a wrong answer after the damage has already been done. The compliance team learns about the issue through a complaint, not a monitoring report. The audit trail for the period of degraded performance is incomplete.

With native performance monitoring, the accuracy drop is detected within the next scheduled evaluation run, before any user is affected. The team is alerted, investigates, remediates, and re-qualifies the agent. The complete event is documented: the detection, the root cause, the remediation steps, and the re-qualification evidence. Nothing needs to be reconstructed.

This is the operational difference between governance that anticipates and governance that reacts.

Continuous performance validation

Always on, from the moment of publish
Agent performance monitoring begins the second an agent goes live.

Evaluations run on configurable schedules: ad hoc for immediate checks, hourly or daily for ongoing operational monitoring. Each run produces scored results across every evaluation dimension relevant to the agent's clinical function. Results are AI-assisted and human-reviewed before they are published and the human is always in the loop.

Configurable thresholds mean the team sets the bar that reflects their clinical context and their compliance requirements. A warning fires when performance drops below the first threshold. A critical alert fires when it drops further. The thresholds are set per dimension, per deployment, so an agent built for a high-stakes clinical workflow can be held to a higher standard than one handling lower-risk queries.

> **I was pleased that the agent was able to validate itself to the 100% target.**
>
> A top-10 pharma

The core dimensions that cover the full behavioral surface

Medable's evaluation framework assesses clinical AI agents across several core dimensions. Together they cover the full surface of agent behavior that matters in a regulated clinical deployment:

01
Response accuracy
Returns accurate, complete answers consistently across clinical workflows.

02
Behavioral consistency
Maintains consistent responses and output patterns. Detects subtle reasoning drift early.

03
Grounded responses
Stays within validated instructions, governed tools, connectors, and controlled knowledge boundaries.

04
Safety
Maintains appropriate scope and content boundaries for regulated clinical use.

05
Tool usage
Calls out to the correct tools, in the correct sequence, for each multi-step clinical workflow.

06
Latency
Responds within operationally acceptable time windows.

07
Task delegation
Leverages connected agents and facilitates downstream workflows as designed.

08
Task completion
Checks against those workflows to ensure the task is completed as specified.

09
User criteria
Configurable and extensible: add organization-specific workflows and compliance requirements.

10
User sentiment
Users can mark if they like the response by using thumbs up or down and provide thoughts on why they provided the thumbs up or thumbs down.

Tailored to your agent, not a generic benchmark

The most important property of Medable's evaluation framework is that it is generated from the agent's actual configuration. The platform looks at the agent's skills, connectors, instructions, and workflows, and builds evaluation probes from that specific context. No external tool has access to this information. No generic benchmark applies.

This matters because a generic evaluation tool assessing a CRA agent does not know what the CRA agent is supposed to do. It can apply general accuracy metrics, but it cannot assess whether the agent is calling the right tools in the right sequence for a site query workflow, or whether it is staying within the knowledge boundaries defined for that specific study. Medable's evaluation probes are specific to what your agent actually does.

This is one of the foundational trust properties of the system: governance that knows the agent it is governing.

QMS-ready reporting by default

Every evaluation run produces a downloadable QMS report. These reports are audit-ready by default, not assembled under pressure when an audit is triggered. Performance history, dimension scores, threshold status, and human-approval records are available at any time from the same system the agent runs in.

This means the evidence of continuous validation is always current, always complete, and always in one place.

Complete, immutable traceability

The compliance gap that versioning closes

Let's imagine for a second that an agent is being used in a clinical trial. Now, let's imagine that the underlying prompt is improved, a connector is updated, and a knowledge source changes. In a regulated environment, every one of these changes needs to be controlled, documented, and traceable. Without versioning, there is no reliable answer to a basic compliance question: what was the agent doing on this specific date, what changed, and who approved it?

That is not a theoretical gap. It is a compliance gap that will surface at the worst possible moment: mid-audit, when the team needs to produce a continuous record and discovers that the evidence requires reconstruction.

How Medable's versioning works

Agent versioning operates on a simple and auditable principle that published agent configurations are immutable. Simply put, once an agent goes live, no one can quietly edit it. Any change to any component creates a new version. The prior version is preserved untouched.

Every component of an agent is versioned independently, including its prompts, skills, connectors, and triggers. Each has its own version history, with full attribution and timestamps. Here, a change to a prompt does not create ambiguity about what else may have changed. The record is precise.

Additionally, users can rollback in one click. If a change creates a problem, the team can return to the last known good version immediately. That rollback itself enters the version history as a new version. History is never

overwritten and the audit trail is never broken.

What this means for regulatory readiness

Immutable version history with full timestamps satisfies computer system validation requirements without a separate validation workflow. The record proves what the agent was configured to do at any point in its deployment, and it proves that no unauthorized changes were made.

Combined with continuous performance monitoring, the versioning record gives the compliance team something more valuable than a point-in-time snapshot: a complete, continuous account of the agent's configuration and behavior across the entire deployment period.

That account does not need to be assembled when an audit arrives. It has been produced automatically, continuously, from the moment the agent was published.

A performance monitoring lifecycle, not a layer

The full loop in one platform

Agent performance monitoring and agent versioning are not two separate tools. They are two parts of a single governed lifecycle that runs without gaps from configuration to ongoing operations. Every stage of that lifecycle produces traceable evidence. There are no disconnected systems as there is no gap between the agent record and the governance record.

Phase	What happens	Evidence produced
Build	Configure skills, connectors, and triggers. Platform auto-generates evaluation probes from agent configuration. Pre-deployment scoring and human sign-off before go-live.	Baseline performance scores, human approval record.
Operate	Scheduled and ad hoc evaluations run automatically. Dimension scores tracked against configurable thresholds. Alerts triggered when performance drops below warning or critical levels.	Timestamped evaluation scores, alert history, QMS reports.
Govern	Every component change creates a new version. Rollback available in one click. Performance reports and version history exportable for QMS filing at any time.	Immutable version history, full audit trail, exportable QMS-ready reports.

Why “built in” is not the same as “integrated”

The distinction between native governance and integrated governance is operational, and matters specifically in regulated environments.

A governance tool integrated from outside the platform has no access to the agent’s actual configuration. It cannot generate evaluation probes tailored to what the agent does. It applies generic benchmarks to a specific, complex clinical workflow. It introduces a new vendor relationship, a new integration surface, and, critically, a gap between the governance record and the agent record.

Native governance knows the agent because it is the same platform as the agent. Evaluation probes are generated from the agent’s actual skills, connectors, and instructions. The audit trail lives in the same system as the version history and there are no seams or gaps where risk can enter.

One system of record

When an auditor asks for evidence, the answer is not spread across multiple systems (“let us pull that from three systems”). It is a single, exportable, timestamped record that has been accumulating continuously since the agent was published.

Again, this is a consequence of the architecture built around governance.

Why building your own AI with performance monitoring does not work

“Big corporations can’t rely on their internal speed to match the transformation that is happening in the world. As soon as I know a competitor has decided to build something itself, I know it has lost.”

CEO of Sanofi

These candid sentences from from Sanofi’s CEO showcase one of the most common questions that are at the forefront of every pharmaceutical company’s mind; whether to build or buy your way into the agentic and generative AI revolutions.

Many organizations consider building their own clinical AI governance capability. The appeal is understandable as they can control the design, remove vendor dependency, and have a system tailored to internal requirements. The reality is more complicated, and the gap between the appeal and the reality tends to surface at the worst possible moment.

In August 2025, MIT reported reported “The data is stark: Only 5% of custom GenAI tools survive the pilot-to-production cliff, while generic chatbots hit 83% adoption for trivial tasks but stall the moment workflows demand context and customization.”

Proofs of concept are still valuable. Experimentation gives teams hands-on insight into what works, where friction arises, and what scaling gaps emerge. It also helps organizations clarify which workflows are truly worth automating. But in clinical research, where stakes are high and competition is intense, learning by doing must not become a multi-year detour.

Testing non-deterministic systems is a specialist capability

Traditional software testing assumes determinism. You define an input, you assert an output, the test passes or fails. That model breaks entirely with AI agents. Because agents are adaptive by design, the same input can produce a different output on every run, not incorrectly, but inherently. There is no test suite that can assert a fixed output from a language model.

The right approach is evaluations rather than tests: scored dimensions rather than pass/fail assertions, AI-assisted judgment rather than rule matching. Building that capability requires deep expertise in evaluation methodology, significant engineering investment, and ongoing calibration as the underlying models change. Most clinical engineering teams have no prior experience building this. It is easy to build something that looks like a governance system but is actually a confidence generator: a tool that reports passing scores while missing the behavioral shifts that matter.

The governance system itself requires validation

In a regulated clinical environment, the governance system is not exempt from validation requirements. It is a software system used in a GxP context. It needs to be validated, its validation needs to be documented, and that documentation needs to be maintained as the system changes.

This applies even when teams build their own governing agents. Evaluation agents can be built to continuously evaluate other AI agents, providing an automated governance layer that measures quality, compliance, and performance against customer-defined criteria. However, that flexibility doesn’t exempt the governance layer itself from scrutiny. If anything, it raises the stakes, since a custom-built evaluator is itself a software system that must be validated.

This is the most common blind spot in build plans. Teams budget for the engineering effort to build an evaluation framework (including their own Evaluation Agents), but they do not budget for the validation workstream that the framework itself requires (the IQ/OQ/PQ protocols, the change control procedures, the re-validation triggered by updates). Without that, the governance system provides no regulatory cover, no matter how sophisticated the evaluation logic is. With it, the team is running a second regulated software lifecycle alongside the agent itself.

Generic observability tools cannot serve a clinical purpose

A number of AI observability and evaluation platforms exist, and some are technically capable. None of them were built for regulated clinical workflows. They do not carry GxP context. They do not produce GxP-formatted reports. Their evaluation dimensions were not designed for the specific behavioral surface of a clinical AI agent.

More fundamentally, a generic tool has no knowledge of your agent’s configuration. It cannot generate evaluation probes from the agent’s actual skills, connectors, and instructions. It applies benchmarks designed for someone else’s use case to your clinical workflow. The coverage gaps are not visible until an auditor finds them.

Integration creates a seam, and seams create risk
When performance validation tooling lives outside the agent platform, there is always a gap between the governance record and the agent record. Version history in one system, evaluation results in another, audit trail correlation done manually. In normal operations, that gap is inconvenient. In an audit, it is a liability.

Auditors in regulated environments expect a continuous, unbroken record. A governance package assembled from exports across disconnected systems raises questions about completeness and chain of custody that a native system never raises. The effort required to assemble that package is also non-trivial, and it tends to be underestimated until the audit is underway.

Governance is a permanent operational commitment, not a project
A built governance system requires a dedicated team to maintain evaluation frameworks as agents evolve, update the system when underlying models change, and keep all validation documentation current. That resource cannot be redeployed to clinical work. It is permanently allocated to governance tooling, for the life of every agent in production.

The total cost of a built governance capability is not the cost of the initial build. It is the cost of the initial build plus the ongoing maintenance plus the validation workstream plus the integration upkeep, compounded across every agent that enters production and every model update that any of them receives.

What this looks like in practice

The architecture argument has practical consequences. These three scenarios illustrate what continuous, native governance makes possible in clinical operations.

Scenario 1

Detecting operational behavior shift before it reaches a site

A CRA-facing agent is deployed to answer site management questions. After a protocol amendment in week eight, response accuracy on enrollment threshold queries begins to decline. The agent's behavior has shifted, not because of a change anyone made to the agent, but because the underlying knowledge base was updated and the model's grounding shifted with it.

The next scheduled evaluation run flags the response accuracy dimension below the warning threshold. The clinical operations team is alerted. They investigate, identify the source of the shift, update the agent's knowledge base configuration, re-run evaluations, and re-qualify the agent before any site coordinator encounters a degraded response. The complete event is documented in the audit trail: the detection, the root cause investigation, the remediation steps, the re-qualification scores, and the human approval of the updated configuration.

Nothing was reconstructed. The evidence was produced automatically as the events occurred.

Scenario 2

Showing up to an audit ready

An audit is triggered mid-trial. The compliance team needs to demonstrate that the protocol review agent was performing within validated parameters for the entirety of its deployment period, and that every configuration change during that period was controlled and documented.

They export the QMS performance report from the platform: continuous evaluation history across all dimensions, threshold status at every evaluation run, version record for every component change with timestamps and attribution, and human-approval records for every evaluation published. All of it from the same system the agent runs in. The evidence is complete without assembly. The audit proceeds.

Scenario 3

A clinical team that trusts what it has deployed

A study team deploys a data query agent. They do not need to wonder whether it is performing correctly between check-ins, because they know the evaluation framework is tailored to the agent's specific workflows and runs on a daily schedule. They know that any significant performance drop will trigger an alert before users are affected. They know that every change they make is versioned and traceable, and that rollback is available if something goes wrong.

They are not managing a governance process alongside the trial. Governance is running. The team can focus on the trial.

That is the operational outcome: not just compliance confidence, but the freedom to deploy with certainty, rather than deploy and hope.

Conclusion

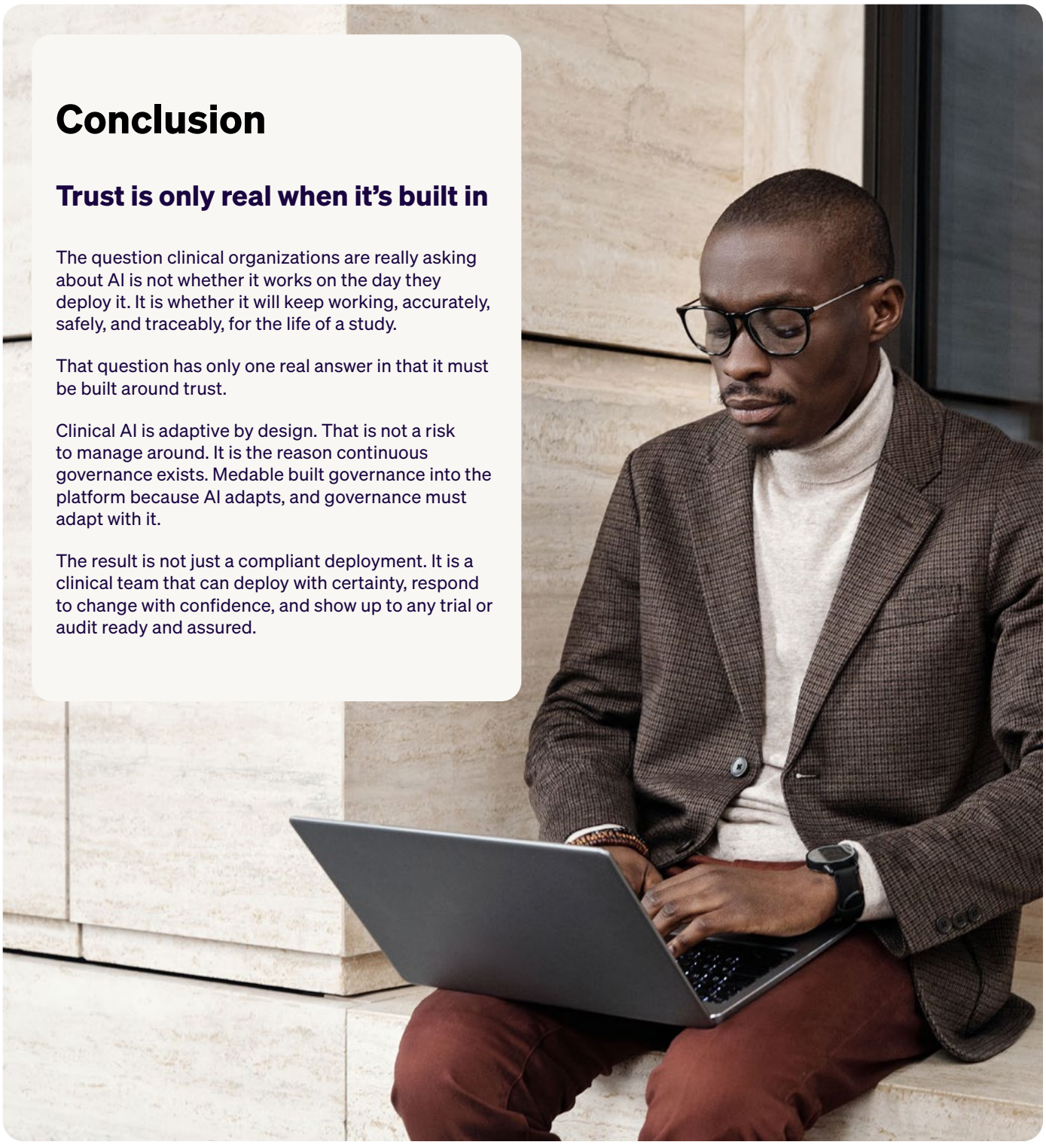
Trust is only real when it's built in

The question clinical organizations are really asking about AI is not whether it works on the day they deploy it. It is whether it will keep working, accurately, safely, and traceably, for the life of a study.

That question has only one real answer in that it must be built around trust.

Clinical AI is adaptive by design. That is not a risk to manage around. It is the reason continuous governance exists. Medable built governance into the platform because AI adapts, and governance must adapt with it.

The result is not just a compliant deployment. It is a clinical team that can deploy with certainty, respond to change with confidence, and show up to any trial or audit ready and assured.



Learn about how Medable's AI-enabled solutions are solving every day trial issues, **here.**

Medable