

Controls on Paper Are Not Controls in Practice

Published: June 2026 • By Ashish Upadhyay, Director Products & Services

Most organizations do not have a controls problem. They have an execution problem.

Policies are documented. Procedures are written. Control matrices are maintained. Training decks are distributed. On paper, the control environment often looks mature.

But the real question is not whether a control exists in a document. The real question is whether the control is structurally enforced when work is actually performed.

A control that depends entirely on people remembering to follow it is not yet a reliable control. It is an intention.

The Illusion of Paper Controls

Many organizations spend significant time defining controls. They identify risks, document expected procedures, assign owners, and describe the evidence that should prove the control was performed.

That work is important. But it is only the beginning. A control documented in a policy, procedure, or spreadsheet does not automatically influence the way work happens.

If the actual process still runs through emails, spreadsheets, shared folders, chat messages, informal approvals, and manual reminders, then the control may exist on paper but not in practice.

This creates a dangerous gap. The organization believes the control is defined, but the process itself may not enforce the behavior the control requires.

Where Controls Break Down

Controls usually fail in the space between documentation and execution.

A procedure may say that evidence must be uploaded before approval, but the tool may allow approval without evidence. A policy may require independent review, but the workflow may allow the preparer to

approve their own work. A control matrix may require downstream validation after a change, but the task tool may not know that downstream work depends on upstream evidence.

In each case, the problem is not lack of documentation. The problem is lack of enforcement.

Common breakdowns include:

- Tasks completed before required evidence is submitted.
- Approvals performed by the same person who prepared the work.
- Required reviews skipped because the workflow does not block progression.
- Completed tasks reopened and changed without downstream revalidation.
- Exceptions captured in comments instead of structured exception records.
- Audit evidence reconstructed later from emails, screenshots, and file timestamps.

When controls are enforced only through reminders, training, or after-the-fact review, the organization is relying on hope as part of its control design.

A Practical Example

Consider a procurement approval workflow. The documented procedure requires three things: the requester uploads supporting evidence, a reviewer validates the evidence, and an approver signs off only after the review is complete.

On paper, this looks controlled.

But if the workflow tool allows the requester to mark the evidence task complete without attaching the required document, allows the same person to perform the review, and allows approval before the review task is finished, then the control is not actually embedded in the process.

The workflow may still show a clean status. The dashboard may still turn green. The process may still be marked complete.

But completion is not the same as control.

The organization may only discover the gap when an auditor asks a simple question: show me how the system prevented this process from moving forward unless the control was followed.

Documentation vs. Executable Control

Documented Control	Executable Control
The policy states what should happen.	The workflow prevents the process from advancing unless it happens.

The procedure describes who should review.	The system enforces who can review and who cannot.
Evidence requirements are listed in a control matrix.	Evidence is required before the task can be completed.
Exceptions are discussed in comments or emails.	Exceptions are structured, attributed, approved, and reportable.
Audit evidence is reconstructed after the fact.	Audit evidence is generated as part of execution.

Why This Matters for Governance

In many organizations, controls are evaluated through periodic testing. A sample is selected. Evidence is requested. Teams search emails, download files, gather screenshots, and explain what happened.

That approach can confirm some activity occurred. But it often does not prove that the process itself enforced the required behavior when the work was performed.

This distinction matters because compliance failures are often not caused by missing policies. They are caused by execution drift. People bypass steps, complete tasks out of order, make exceptions informally, or correct mistakes without preserving the original record.

A stronger control environment reduces that drift by designing controls into the workflow itself.

What Controls in Practice Look Like

Controls in practice are not just written requirements. They are operational rules built into the way work gets done.

A governed workflow should be able to enforce and prove:

- Required evidence was submitted before approval.
- The preparer and approver were different people where separation is required.
- Tasks occurred in the required sequence.
- Required reviews could not be skipped.
- Reopened or changed tasks triggered revalidation where needed.
- Exceptions were captured with owner, reason, approval, and timestamp.
- The audit trail shows not just activity, but control performance.

This is the difference between a control that is documented and a control that is executed.

The Shift to Governed Workflows

Organizations do not need more policy documents to solve this problem. They need workflows that convert policy intent into process behavior.

That means translating controls into rules, dependencies, required evidence, approval conditions, separation of duties, exception paths, and audit-ready records.

When this happens, governance becomes part of the execution layer. The workflow does not merely record what people say they did. It guides, constrains, validates, and preserves the work as it happens.

This does not mean workflows should become slow or overly bureaucratic. Well-designed governance should make the right path clear and the risky path visible. It should allow exceptions when needed, but ensure those exceptions are explained, approved, and traceable.

The Real Question

If your organization has documented controls, that is good. But it is not enough.

The next question is whether those controls are actually embedded in the daily systems where work is performed.

Can the workflow enforce the required sequence? Can it prevent conflicting actions? Can it require evidence before approval? Can it preserve what changed and why? Can it prove the process was followed without asking someone to reconstruct it later?

Controls on paper describe the intended process. Controls in practice shape the actual process.

Because a control that only exists in a document may satisfy a checklist. But a control that is built into the workflow protects the integrity of how work actually happens.

Ready to move from documented controls to executable controls?

Candent helps organizations design governed workflow solutions where approvals, evidence, sequencing, exceptions, segregation of duties, and audit-ready history are built into the process - not reconstructed after the fact.

Contact Candent today to strengthen the way your controls operate in practice.