

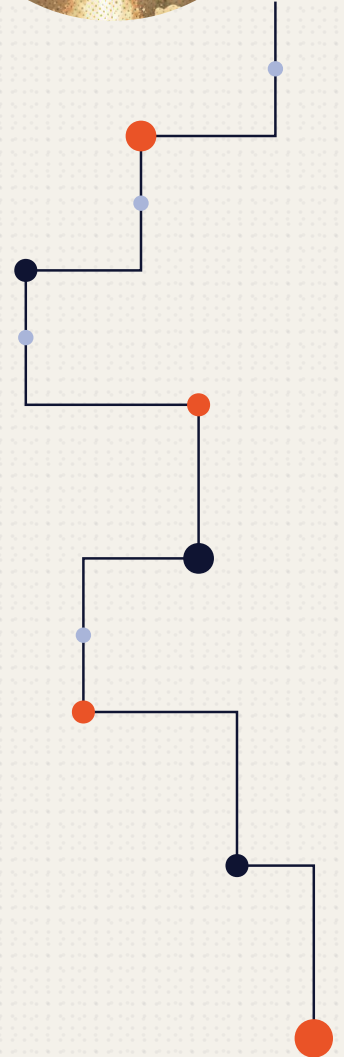
Dispelling the coverage illusion

A QA leader's playbook for end-to-end SAP quality in the agentic era.



Contents

- 03** **The coverage illusion**
- 04** **Three shifts to reshape how SAP quality is measured**
- 04** **Shift 1 – From test case to business process**
- 04** **Shift 2 – The test pyramid inverts**
- 05** **Shift 3 – QA moves from authoring to governing**
- 05** **What role agentic AI really plays**
- 06** **Four agentic capabilities for SAP software testing**
- 07** **A day in the life with Test Cloud**
- 08** **The playbook – five moves worth making now**
- 09** **Go deeper**



SECTION 1, THE COVERAGE ILLUSION

The coverage illusion

SAP testing teams are tracking a number that quietly stopped meaning what it used to: coverage. The percentage of transactions, screens, or test cases exercised has been the default QA scorecard for 25 years. But coverage measures what's easy to count, not what's likely to break. And in modern SAP landscapes, the things most likely to break don't live only inside SAP; they live at the seams and in the in-between.

Coverage is a vanity metric inherited from a world where SAP was the entire business process. That world is gone, and it didn't drift away — it was pushed.

Three forces dismantled it. First, SAP's own clean-core strategy moves custom logic out of the core and into side-by-side extensions, SAP BTP services, and API integrations — by design. Second, SaaS systems have swallowed entire functional domains that used to live inside SAP: Salesforce for CRM, ServiceNow for ITSM, Workday for HR, headless stacks for commerce. Third, release-cadence acceleration on every side: SAP's quarterly rhythm, vendors shipping weekly, and integration platforms updating in the middle of it all.

The volume of change flowing through the average enterprise per quarter — or even per day — is an order of magnitude higher than a decade ago, and very little of it lives inside SAP.

The result is what every QA lead already sees: a typical order-to-cash, procure-to-pay, or close-to-report cycle lives outside and between SAP and other applications — in Salesforce, ServiceNow, custom approval apps,

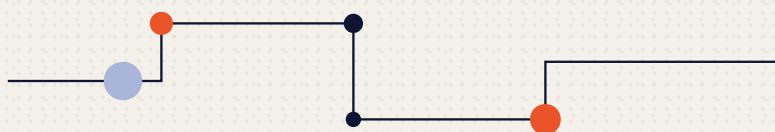
middleware, third-party portals, and the integration glue between them. Yet most SAP QA measurements still live inside the SAP boundary.

Across enterprise SAP landscapes, the patterns that cause late-release pain tend to look similar: integration handoffs that pass QA individually but time out in combination; pricing or approval edge cases that span SAP and a custom app; data transformations in middleware that silently mutate on a release; SAP Fiori UI drift after a quarterly release that breaks a downstream flow; authorization changes that don't fail, but produce wrong results invisibly.

Few of these are findable through SAP-bounded test cases. A 100% SAP coverage report tells you little about any of these scenarios.

The solution isn't to add more coverage, but to change the unit of measurement. The question QA needs to answer stopped being "did the transaction pass?" years ago. Every other metric is a proxy, and most of the proxies we've been using are broken.

Coverage measures what's easy to count, not what's likely to break.



SECTION 2, THREE SHIFTS TO RESHAPE HOW SAP QUALITY IS MEASURED

Three shifts to reshape how **SAP quality** is measured

So, "coverage" has stopped meaning what it once used to. Now what? Ultimately, quality is being simplified to this core metric: a business process completes correctly, end-to-end, for the customers and workflows it was supposed to serve. But meeting that standard requires three major shifts in your SAP quality assurance strategy.

SHIFT 01

From the test case to the **business process**.

Test case inventories were always a representation for effort, but not necessarily for risk. In our agentic era of software testing, the unit of reporting should match the unit of business impact.

That means retiring the "tests passed / tests failed" slide in your release review and replacing it with integrity scores by

process: Order-to-Cash, Procure-to-Pay, Close-to-Report, Hire-to-Retire. Each process carries a score tied to the outcomes the business actually cares about — fulfillment time, cash cycle, close accuracy, onboarding speed. When the score drops, QA teams produce a diagnosis, not a test count.

SHIFT 02

The test pyramid **inverts**.

For a quarter century, QA has taught that unit tests are plentiful and cheap, integration tests sparse and moderate, and end-to-end tests rare and expensive. That pyramid reflected the cost curve of manual authoring — not what actually delivers the most risk reduction per dollar.

Today, agentic software testing is bending the cost curve. Generation from real usage traces, self-healing across systems, and orchestrated runs spanning SAP and non-SAP tools make end-to-end tests the newly cheap work. What becomes scarce is human exploration and context — the judgment call about which new scenarios are worth testing,

which risks are acceptable, and where the model is wrong about what matters.

The new pyramid has a broad base of auto-generated, self-healing, process-level coverage; a smaller band of integration tests above it; and, at the top, human-driven exploration that finds what agents won't predict. If your strategy still treats regression as the expensive tier, you're about to be out-executed by teams whose regression is automatic and whose scarce human hours go to the work that prevents the next production incident.

SHIFT 03

From authoring to governing.

The inversion of the software testing pyramid naturally means the role of the people doing the testing will change too. The high-leverage QA leader of the next three years isn't a tool expert — agentic software testing solutions will generate the tests, execute them, and analyze the results.

Instead, testers become risk-portfolio managers — deciding which processes need continuous green, which yellow is tolerable, which integrations warrant hourly

surveillance, and which can ride on daily checks. They set policy, govern exceptions, and own the integrity scores. This is the role QA has been quietly evolving toward for a decade; agentic AI is forcing the issue.

Taken together, these three shifts add up to a real change in not only how QA is done, but what it's really for. Not simply faster testing, but fundamentally different testing: measured differently, staffed differently, and reported differently.

Not simply faster testing, but fundamentally different testing.

SECTION 3, WHAT ROLE AGENTIC AI REALLY PLAYS

What role agentic AI really plays.

Most "AI in testing" stories describe AI making the existing practice faster or easier. Agentic AI is a different kind of shift — changing which practices are worth doing at all.

That's a stark claim, so we'll dive into the specifics. The prior generation of AI testing tools — record-and-replay

assistants, flaky-test detectors, auto-selectors — made manual testing less painful. Agentic AI removes categories of work entirely. In the context of end-to-end SAP quality, four agentic AI capabilities carry most of the weight.

Four agentic capabilities for SAP software testing.

● 01 • Test generation from real process traces

Agents read documented requirements and production usage data — not just screens. The tests they generate reflect the paths the business truly walks, including the non-SAP segments no traditional SAP tool sees.

● 02 • Change impact traced through the landscape

When a transport is staged, agents correlate it with downstream objects, integrations, and usage patterns to surface a risk-ranked list of processes to re-validate. Over-testing (expensive) and under-testing (catastrophic) both go away, replaced by a prioritized plan that fits the release window.

● 03 • Healing that repairs across every layer

The right agentic testing solution covers SAP S/4HANA, SAP ECC, SAP Fiori, SAP WinGUI, SAP WebGUI, and SAP Business Client — plus web, mobile, API, and legacy desktop. Self-healing technologies repair broken selectors in real time wherever they drift, inside SAP and in whatever application screen follows it.

● 04 • Orchestration of the full process

Coordinate AI agents, automations, and humans into one testing flow. A regression for a pricing change runs SAP transport validation, a Fiori approval path, a custom pricing app, a Salesforce sync, and a downstream invoice check as a single orchestrated pipeline — returning a process-level verdict, not seven disconnected tool reports.

Enter UiPath Test Cloud.

And a new generation of agentic software testing tools has arrived in [UiPath Test Cloud](#). It includes all of the features listed above, and integrates with SAP Cloud ALM. Test Cloud facilitates this operating-model change, and slots it into your existing SAP delivery governance. No parallel processes to build. No new compliance workstream. Role-based access, auditability, and production-grade security align to enterprise expectations from day one.

SECTION 4, A DAY IN THE LIFE WITH TEST CLOUD

A day in the life with Test Cloud — measured by **processes**, not tests.

So what does this new QA philosophy, paired with an agentic software testing tool, look like in practice? A Tuesday release into a regulated SAP landscape, reframed around outcomes rather than tests.

Transports stage Monday evening. Overnight, agents correlate the contents with production usage and integration maps. By Tuesday morning, the QA lead opens Test Manager in Test Cloud and sees a risk-ranked set of business processes: Order-to-Cash elevated due to a pricing logic change; Close-to-Report unchanged; Procure-to-Pay slightly elevated. The lead approves the plan, adjusts priorities for a business-critical scenario, and kicks off execution. Maestro runs SAP automation, SAP Fiori flows, a Salesforce handoff,

an API check, and a business-user review as one coordinated pipeline. Mid-run, an SAP Fiori selector shifts; the Healing Agent repairs it in real time.

By late morning, the release meeting has a process-level verdict: Order-to-Cash green, Procure-to-Pay yellow with a diagnosed root cause in a Salesforce sync, Close-to-Report green. The QA lead isn't defending a coverage percentage, they're explaining a risk decision.

This isn't hypothetical.

Enterprises running SAP at scale are already operating this way. Cisco cut manual testing time by 50%^[1], NatWest lowered test maintenance by 75%^[2], and EDF extended automation across end-to-end processes^[3] by running them as orchestrated flows rather than testing more of any one system.

50%

Cisco

Reduction in manual testing time^[1]

75%

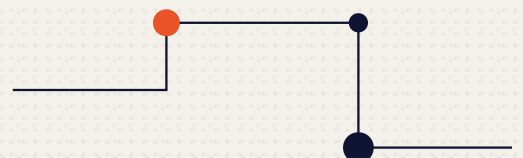
NatWest

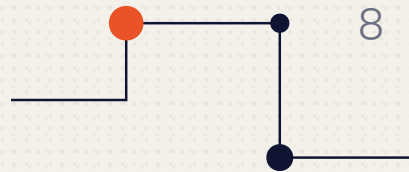
Lower test maintenance^[2]

E2E

EDF

Extended automation across end-to-end processes^[3]





The **playbook** — five moves worth making now.

If this all makes sense to you (and you're keen to get started) the question is what to do about it tomorrow. Five moves worth making now — whether you're ready to adopt new tooling tomorrow.

1**Retire coverage percentage as your primary KPI.**

Replace it with process-integrity scores by business domain. Report them to the same level of the organization that reads the financial close.

2**Audit where your failures actually originate.**

In many enterprises, a sizable share of SAP-linked production defects manifest outside SAP. Rebalance your testing investment to where the risk lives, not where the tool has always pointed.

3**Push end-to-end runs earlier.**

They're newly cheap. Make them the regression floor — not a pre-release event held once and hoped over.

4**Invest in exploration as a distinct discipline.**

Train senior testers to hunt what agents won't predict. That's where next-release surprises hide, and it's the work humans are uniquely good at.

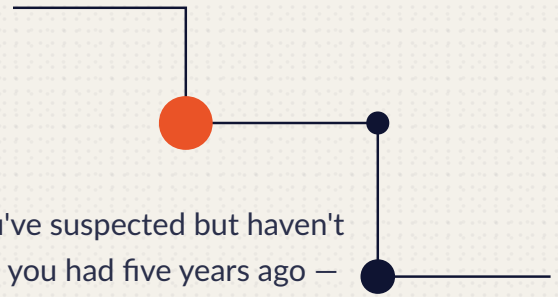
5**Rebuild your release meeting around process verdicts.**

Kill the pass-rate slide. Process-level green/yellow/red tied to business impact is the format that earns QA a seat at the release decision table — and keeps it.

SECTION 6, GO DEEPER

Go deeper.

If any of this lands — if the coverage illusion is something you've suspected but haven't named, or if you've been staffing QA for the operating model you had five years ago — there's more to explore.



Explore SAP testing with UiPath Test Cloud.

Visit our SAP testing hub to see UiPath Test Cloud for SAP in action, find upcoming webinars and workshops on end-to-end software quality, and get a practical framework for moving your team past coverage and into process integrity.

[Visit the SAP testing hub →](#)

REFERENCES · [1] CISCO, UIPATH CUSTOMER OUTCOME — 50% REDUCTION IN MANUAL TESTING TIME. · [2] NATWEST, UIPATH CUSTOMER OUTCOME — 75% LOWER TEST MAINTENANCE. · [3] EDF, UIPATH CUSTOMER OUTCOME — EXPANDED AUTOMATION COVERAGE ACROSS END-TO-END PROCESSES.