Enterprise agentic processes and tomorrow's interfaces: what software engineering puts back at the centre · Junyr Method™ · September 2026 Edition

Paul-Antoine TUAL

September 2026

Contents

> ***The thesis.*** *While AI mainly produced answers, conversation could be enough; once it executes a process, the organisation must control decisions, authorisations, resumptions and evidence of execution.*
>
> - *The decision authorising an artefact becomes as important as producing it.*
> - *Reliability depends on the complete sequence, its controls and its dependencies.*
> - *The useful interface becomes a shared control room rather than a simple chat history.*

1. The bottleneck is now the decision

Models that drafted answers can now read a database, call an API, prepare a document or trigger a notification, so value, risk and interface requirements now also concern execution.

- **Value**: when an agent produces forty follow-up messages in a minute, reading each message scales poorly, while approving the follow-up plan remains workable.
- **Reliability**: output quality depends on the model as well as the order of the steps, controls, resumptions and permissions.
- **Interface**: the user needs to see what awaits a decision, what happened and where to take back control.

An internal register covering seventeen days illustrates this movement towards upstream decisions, without constituting a measurement that can be generalised to other teams or projects.

- **Observed volume**: 565 plans written and 477 delivered.
- **Documented stops**: 31 plans abandoned, including 26 before a line of code was written.
- **Interpretation**: these 26 stops represent work avoided after clarification, rather than a universal return rate (*Decision-gated development*).

2. Skill and workflow: two objects, two promises

A skill captures know-how left to the model's judgement, while a workflow puts the sequence into versioned code; the two objects can complement each other, but they do not promise the same repeatability.

- **Skill**: a folder organised around a `SKILL.md` file whose header requires a name and a description.
- **Skill portability**: the format was launched on 16 October 2025, published as an open standard on 18 December 2025, then documented by OpenAI, Microsoft and Google; the standard's directory listed 46 compatible products on 30 August 2026.
- **Skill promise**: reusable know-how that the business can revise, with triggering and outputs that remain nondeterministic.
- **Agentic workflow**: loops, filters, joins, conditions and retries written once, replayable and testable.

- **Workflow promise**: the Claude Code documentation distinguishes the “instructions” reused by a skill from “the orchestration itself” reused by a workflow.

The Agent Skills testing procedure measures several runs of the same case because triggering depends on the model, and this trial protocol does not describe successive stages of a workflow.

- The official procedure asks for three trials per case.
- It accepts a trigger rate above 0.5 as the criterion for the description under test.
- This rate describes the observed frequency of triggering; it guarantees neither an identical output nor the success of a complete business chain.

In the Junyr Method™, agents are used first for operations where judgement over unstructured content adds value that deterministic code does not directly provide.

- **Analyse**: extract useful information from a document, message or corpus.
- **Synthesise**: reconcile divergent sources and rank the useful elements.
- **Present**: adapt an output to a recipient, format and context.
- **Code the deterministic work**: count, sort, join, cap or write to a database when the rule admits a verifiable answer (*Enterprise agentic processes*).

3. Chain reliability depends on conditional probabilities

The product $0.95^{20} \approx 35.8\%$ and the result $0.95^{3} \approx 85.7\%$ illustrate only a model in which twenty or three stages each succeed 95% of the time and their outcomes are independent.

- **Under independence**: reducing the number of required stages changes the illustrated result from 35.8% to 85.7%, without changing the model executing each stage.
- **Without a dependence assumption**: twenty events each succeeding 95% of the time can jointly succeed anywhere from 0% to 95%; for three events, the range is 85% to 95%.
- **In a real workflow**: end-to-end probability is calculated with conditional probabilities because a stage can depend on previous outputs, errors or retries.
- **Design consequence**: removing an unnecessary stage reduces exposure to an additional failure, while adding validation or a retry can improve overall system reliability.

Toolathlon documents correlation across repeated trials of the same benchmark and must not be treated as an experiment involving different stages of a workflow.

- The benchmark contains 108 tasks exposing more than 600 tools across 32 applications.
- The best model reported in July 2026 succeeds on **80.6%** of tasks on the first trial and **73.1%** three times in a row.
- If the three trials were independent and shared the 80.6% marginal rate, their product would be 52.4%, below the observed 73.1%.

- This gap shows dependence between repeated trials in this benchmark; it provides neither a universal lower bound nor a success rate for process stages (Anthropic, *System Card: Claude Opus 5*, 24 July 2026, §8.13.6).

AutomationBench provides a separate measurement of long business-flow execution in a simulated company, on a scope that is neither Toolathlon nor the production environment of a particular company.

- AutomationBench, published on 21 April 2026 by two Zapier researchers, covers 47 applications and scenarios inspired by customer flows in sales, marketing, operations, support, finance and human resources.
- The April paper places the best models below 10% full completion on this benchmark.
- The July reading in the *System Card: Claude Opus 5* (§8.13.7) reports **26.0%** for the best model evaluated.
- A careful reading is that approximately three quarters of the benchmark scenarios remain incomplete in that configuration, supporting supervision without turning this score into a per-client process probability.

4. Four software engineering practices structure a sustainable process

Four software engineering practices form a useful foundation for agentic processes, with their selection and implementation depending on the business, data and risk level.

- **Declare idempotence** to distinguish a safe retry from repetition that duplicates or destroys an effect.
- **Validate outputs by machine** when the structure or rules can be checked.
- **Build evaluations upfront** to measure regressions against reference cases.
- **Write human decisions and the audit trail into code** so the agent cannot bypass the policy.

The MCP protocol provides a useful vocabulary for describing tools while making clear that annotations declared by a server do not replace a deterministic security barrier.

- The official MCP schema defines read-only, destructive, idempotent and open-world annotations.
- An undocumented tool is treated by default as destructive and non-idempotent.
- The specification says these annotations should not be trusted outside a trusted server.
- The maintainers note that a server can advertise an operation as read-only while deleting files, which creates the need for an independent control.

Structured outputs and evaluations give the system executable checkpoints, provided that reference cases are budgeted and structural conformity is not confused with content accuracy.

- MCP lets a server publish an output schema with which the result must conform and lets the client validate it.
- For batch, destructive or high-stakes operations, Anthropic recommends analysing, producing a plan, validating that plan with a script, executing and then verifying.
- The same documentation describes evaluations as a source of truth while noting the lack of a built-in mechanism to run them.

- A project without a reference case set can observe a visible outage but struggles to measure gradual quality degradation.

Human approval becomes an effective control when it covers the action and its inputs, expires and leaves a record that supports audit or resumption.

- The MCP revision of 28 July 2026 requires that a human can refuse a tool invocation.
- It asks clients to display inputs before the call, confirm sensitive operations, apply timeouts and log execution.
- The 2026 OWASP Top 10, based on 7,714 incidents of which 6,639 were classified, puts excessive agency in third place.
- Since 29 April 2024, ANSSI has recommended prohibiting automated use for critical actions and limiting actions triggered by uncontrolled inputs.

Across 240 attacks, NetInjectBench illustrates that a metadata policy can preserve utility better than a static allowlist, without showing that any defence will reproduce these rates outside the experimental protocol.

Arrangement evaluated on NetInjectBench	**Unsafe tool actions**	**Utility preserved**
Naive execution	82.50%	Baseline
Four prompt-level defences	25.63% down to 10.00%	Baseline
Static allowlist	5.00%	**0%**: every approved change is blocked
Metadata policy barrier	**0 out of 240 benchmark cases**	99.17% and 100%

5. Conversation cannot carry an enterprise process

Conversational interfaces remain effective for exploration, drafting and learning, but a durable process requires infrastructure stability and shared state that chat history does not provide on its own.

- **Harness**: the software around the model changes across releases and can alter observable behaviour.
- **Model**: deprecation policies require planning for migration and regressions.
- **Connectors**: third parties administer them and available tools may vary from one session to another (*Agentic design*).
- **Business state**: a conversation is not, by default, the shared source of decisions and execution.

A working interface must expose four states that often remain implicit or scattered in a conversation.

- What awaits a **decision**.
- What **failed** and at which stage.
- What constitutes an **admissible record**.
- What a **third party can resume** (*Conversation cannot become a working interface*).

6. The scattering of surfaces predates agents and is made worse by them

Agents often add a new surface to decisions already divided between messaging and business tools, making consolidation of state more urgent than the creation of another channel.

- **Internal observation**: an analysis of 874 business conversations estimates that two thirds of so-called useful email could be replaced by better-designed surfaces (*The end of email and Office suites by 2030*).
- **Agentic effect**: an additional chat window removes neither the inbox nor the business tool.
- **Operational risk**: no single surface then holds the complete state of approvals, errors and resumptions.

7. The control room rests on four properties

A useful control room brings decisions, authorisations, events and resumptions into a common state, as illustrated by our internal replacement of Git with an intention register containing two tables, six states and one human approval gate (*We removed Git*).

- **Decision queue**: an ordered, shared view of what awaits arbitration.
- **Authorisation register**: what agents may do, who authorised it and which control checks execution.
- **Append-only log**: a dated history that preserves events instead of silently rewriting the past.
- **Explicit resumption**: the interruption stage, preserved state and next authorised action.

The maturity test is whether the organisation can answer a governance question without searching through chat history.

- Where is the list of active authorisations?
- Who approved each one?
- Which control verifies that execution complies with the authorisation?
- How does a third party resume an interrupted process?

8. The generative interface: one design proposal, three limits

A design study carried out for our work proposes a strategic execution platform with four zones — multi-scale control, workflow canvas, governance and supervision — whose status remains a design hypothesis without usage measurement or experimental validation.

- **Abstraction continuum**: show the strategic option, roadmap and execution graph at three levels of detail suited to leaders and teams.
- **Plan-execution comparison**: overlay the intended path and the path actually taken so deviations can be inspected.

- **Approval card**: display the intent, risk level, exact action, technical detail and fallback plan.
- **Four-zone organisation**: bring multi-scale monitoring, the graph, governance rules and alerts together without claiming this layout is the only possible one.

The proposal becomes operational only after addressing three limitations that concern both ergonomics and architecture.

- **No measurement**: no user test, cost or comparison establishes its superiority.
- **Maturity prerequisite**: the process must already be expressed as a graph.
- **Passive approval**: the summary must not hide the exact action, and approval volume must remain compatible with human attention under OWASP guidance.

9. The approval rule belongs to the code

Reliable approval must be enforced by program logic according to the risk and reversibility of the action, so the agent can neither skip it nor redefine its own scope.

- **Recoverable action**: a graduated policy can permit automated execution with logging and rollback.
- **Irreversible or customer-visible action**: the logic routes the proposal to human review before commitment.
- **Complete mediation**: OWASP recommends applying authorisation to every relevant access in the program.
- **Implementation**: the guardrail belongs in code and can be tested independently of the instruction (*Engineering agentic systems*).

10. Three tiers of use, five conditions for moving up

The Junyr Method™ usage scale describes three levels according to what the machine executes alone and when a person commits the company, without assuming that every process should reach bounded autonomy.

Tier	Operating mode	Cross-functional examples
1. Assisted	AI produces on instruction, the human drives every stage	Drafting a proposal, analysing a file, first version of a deck
2. Supervised	The agent executes inside information-system tools, with human approval before any committing action	Quotation agent, customer-reply agent, reporting agent: nothing leaves without sign-off
3. Bounded autonomy	Workflows, skills and scheduled tasks execute under guardrails and logging	Daily monitoring, scheduled follow-ups, periodic summaries

Moving to the third tier is a method decision that requires five cumulative conditions to be verified within the relevant scope.

- An **identified human decision-maker** for each scope of commitment.

- **Artefact review replaced where possible** by executable checks, with human review retained where judgement remains necessary.
- A **staging area** and proven rollback path.
- A **verification net** tested on representative data.
- **Tested backups and restores**.

When a condition is missing, the supervised tier remains prudent, and some high-stakes processes may remain there indefinitely.

- Autonomy is not a mandatory objective.
- The tier depends on impact, reversibility and control quality.
- A decision not to automate an action can be a valid result of scoping.

11. Eleven software engineering disciplines offer useful analogies

Software engineering disciplines provide a design vocabulary for agentic processes, with each transposition adapted to its context and checked against its actual effects.

Software engineering discipline	Transposition to the agentic process	Example outside code
Specification	Decision-complete plan before execution: scope, contracts, out of scope	Campaign brief signed before content generation
Code review	Review the decision upstream when reading every artefact no longer scales	Approve the follow-up plan before producing forty messages
Version control	Intention register: who authorised what, when	Register of authorisations granted to sales agents
Branching and merging	Parallelise preparation, serialise commitment	Two plans prepared in parallel, one commitment sent to the customer
Testing and continuous integration	Executable checks independent of the agent	Checks on a quotation: margin, legal notices, VAT
Deployment and rollback	Staging area, human promotion, planned cancellation	Drafts on hold, sending after approval, known cancellation procedure
Observability	Append-only log and dated history of decisions	Traceability of actions executed by agents
Post-mortem	Feed incidents back into process design	Customer complaint turned into a new control rule
Technical debt	Process and data debt	A duplicate-ridden repository propagates errors into agent outputs
Resource management	Context and token management	Targeted retrieval avoids rereading the entire history on every run
Distributed architecture	Sub-agents, checkpoints and resumption	Campaign divided into sourcing, qualification and drafting, with each batch checked

The 2026 ecosystem borrows this vocabulary while leaving implementation responsible for the effective guarantees.

- MCP standardises an idempotency indicator, then makes clear that it remains declarative.
- The OpenTelemetry conventions for agents, added on 5 May 2026, still carry *Development* status and should be evaluated accordingly.
- *12-Factor Agents* offers “own your control flow” as a design principle, but its editorial activity and project status do not make it a standard.

12. Support runs on two tracks and at four cadences

The Junyr Method™ proposes a common start — scoping with leadership followed by work on real data with teams — before the strategic governance and operational-support tracks evolve separately.

- **Diagnostic morning**: scope, risks, expected outcomes and decisions with leadership.
- **Diagnostic afternoon**: practical work, friction points and real data with teams.
- **Shared deliverable**: a roadmap connecting leadership decisions to constraints observed in execution.

The two tracks serve distinct responsibilities and can therefore run at different cadences without contradiction.

- **Monthly strategic track**: leadership review, scope arbitration, prioritisation of the next work package, investment and written record.
- **Regular operational track**: unblocking, review of implementation choices and course correction with teams.
- **Operational cadences**: two sessions a month to install the method, one a week at a steady pace, two a week during intensive construction, or no imposed frequency.
- **Cadence review**: a decision at month end according to progress and need.

A Deloitte study dated 11 August 2026 describes a business-process readiness gap in a sample of 501 executives whose organisations were already piloting at least one agentic solution.

- **Reported adoption**: 15% had reached orchestrated multi-agent adoption at scale within this sample.
- **Process dimension**: 21% readiness, last among the seven dimensions measured.
- **Causes cited**: poorly documented or understood processes, fragmented data and systems, and entrenched working habits.
- **Scope**: the findings describe the study sample and framework; they support prioritising process work without establishing an architecture law (Deloitte).

13. The first deliverable takes the form of a numbered list

The first deliverable describes the stages of a real process and asks three questions about each stage to separate deterministic rules, useful judgement and commitments requiring approval.

- **Verifiable answer**: if the stage admits one checkable answer, put it in code.
- **Content judgement**: if it requires reading unstructured material, reconciling sources or adapting a presentation, evaluate the value of an agent.
- **Management decision**: if the rule can be settled once, decide it upstream and version it.
- **Committing action**: if the output is irreversible or customer-visible, enforce human approval in code.

The first process benefits from meeting four cumulative criteria, with a minimum frequency aligned with the proposed examples.

- Already **written down**.
- Run **regularly, at least once a month**.
- **Measurable** in time spent or outcome.
- **Without an irreversible customer effect** in its first version.

Three common candidates make this framing concrete while covering different cadences and functions.

- Chasing quotations left unanswered.
- Triaging inbound requests.
- Preparing a monthly committee meeting.

For a simple, already documented case, half a day may be enough to produce this first map without buying a tool, but the actual time depends on complexity, participants and the quality of existing documentation.

- **Output**: an ordered list of stages, decisions, inputs and effects.
- **Reading**: an estimate of the number of nondeterministic calls and of the rules that can become deterministic.
- **Limit**: the map opens discussion and testing; it does not settle feasibility or risk on its own.

14. Three objections we take seriously

Three objections define the method's boundary: the path can emerge at run time, a format constraint can degrade content, and long-horizon benchmarks can call for better models rather than an external skeleton.

- **Dynamic graph**: Inngest distinguishes a workflow known at design time from the path selected at run time by the model; the useful guarantee then concerns state, boundaries and resumption even when the full graph is not known upfront.
- **Late constraint**: a 20 May 2026 study covering 15,000 generations by small models raises structural validity from 61.5% to 100% under a hard schema while answer accuracy falls from 19.7% to 11.0%; the result supports validating reasoning and format separately without extrapolating to all models.
- **Long-horizon tasks**: OSWorld 2.0, published on 28 June 2026, reports 20.6% full completion and a 54.8% partial score across 108 workflows with a median human duration of one hour and thirty-six minutes; the authors emphasise agent endurance, while our recommendation adds external checkpoints today.

Formalisation turns a vague intuition into testable decisions about scope, rules, data and authorised effects, including cases where the conclusion is not to automate.

- It reveals the stages governed by a stable rule.
- It isolates those where model judgement merits evaluation.
- It identifies the actions that should remain under human decision.

First published on 30 August 2026; revised on 6 September 2026. See also AI Maturity of French SMEs and From PoC to Industrialisation.