VERIFICATION & EVALUATION

The reliability level. A black-box deployment gate.

Measuring whether an AI is right is harder than it sounds, and the obvious methods quietly lie. This is the method that turns it into a number you can defend: self-consistency sampling plus conformal calibration, one reliability level per system and task, with a finite-sample, distribution-free guarantee that holds regardless of the model's bias. For the people who have to defend the bar at which a system earns the right to deploy.

FOR

Head of AI Ops, technical reviewers, model risk, audit

METHOD

TrustGate. Conformal + self-consistency. Open source.

SOURCE PAPER

Mouzouni, 2026. Preprint. github.com/Cohorte-ai/trustgate

DIRECT LINE

charafeddine@cohorte.co

02 THE PROBLEM

A 94%-accurate system is not a 94%-trustworthy system.

Accuracy is what happened on a test set. Trustworthiness is what will happen at deployment, with a quantified confidence guarantee, on inputs that resemble the ones the system will actually see. The distinction is not academic. It is the difference between a paper that gets published and a system that survives audit.

Four families of evaluation are in common use. Each captures something; none survives a deployment gate, and each fails in its own way.

VARIANCE · NOT REPRODUCIBLE

The vibes check

Ten examples looked good, so it shipped. Real signal in a sketch session; incommunicable to a regulator, and a different answer every time you run it.

WRONG DISTRIBUTION

The benchmark score

MMLU, TruthfulQA, the suite of the quarter. A real number, but for the public test set, not for your workflow's actual inputs.

AMPLIFIES BIAS

Naïve majority vote

Agreement carries information, but a confidently wrong model returns the same wrong answer every time, and the vote crowns it.

IMPORTS BIAS

LLM-as-a-judge

A second model grades the first, riding its own position, verbosity and self-preference biases (Zheng et al., 2023) on top of the agent's errors.

> Plausible and correct are not the same thing. The hard cases are the ones that look right.

03 THE EVALUATION STACK

Evaluation is a stack, not a single number.

Accuracy is one number on a past test set. Real evaluation is layered, and each layer asks a different question. Most teams build the first layer and stop; the failures that survive into production live in the layers they skipped.

DETERMINISTIC GRADERS

Unit evals

Small behaviours, checked exactly. Does the tool-call match the schema? Does it refuse the forbidden action? Does it cite only the documents it retrieved? Graded by code, not opinion.

REALISTIC TASKS

Scenario evals

A whole task with a known answer: book the meeting under three constraints; handle the email carrying a prompt injection. Each has a start state, a goal, allowed and forbidden actions, and a rubric.

THE PATH, NOT THE ANSWER

Trajectory evals

For an agent the final answer is not enough. Did it pick the right tools, in the right order? Did it notice a failed call and recover? Did it escalate when it should have?

ONLINE, FOREVER

Production evals

Real users generate cases your benchmark never imagined. Sampled human review, drift detection, tool-call anomalies, regression by capability. Offline evals are necessary and never sufficient.

These four are the basics every team should run. The rest of this briefing makes the hardest part rigorous: turning a layer's "it passed" into a reliability level with a finite-sample guarantee, so the bar is a number you can defend, not a verdict you have to trust.

04 THE RELIABILITY LEVEL

α One number per (system, task).

Distribution-free. Finite-sample. Calibrated.

The reliability level is a confidence score that satisfies three conditions. It is **distribution-free**: no assumption is required about the input distribution the system will see, beyond the exchangeability of the calibration set with deployment. It is **finite-sample**: the guarantee holds for any calibration set size, with the slack characterised by 1/(n+1). It is **black-box**: it requires no access to model weights, logits, or gradients. The system can be a closed API, a hosted endpoint, a vendor-supplied agent.

The construction is the composition of two ideas that pre-date LLMs. **Self-consistency sampling** (the model is queried multiple times; the agreement profile is recorded) provides the empirical signal. **Conformal calibration** (Vovk, Gammerman, Shafer, 2005) translates the empirical signal into a confidence guarantee with a known coverage property. The composition produces the reliability level: an upper bound on the probability that the system returns an incorrect answer on a fresh input from the same distribution, at the declared confidence.

The formal statement is on the next page in the side panel. The intuition is sharper: the system is asked to commit. When the agreement profile is tight, the commitment is narrow and the confidence is high. When the agreement profile is broad, the commitment is wide and the system is allowed to admit it does not know. This is the asymmetry of honesty restored to the system: the model never says *I do not know*; the wrapper around the model can.

DEFINITION

The **reliability level α** of system S on task T is the smallest value such that the probability of S returning a correct answer on a fresh input from T's distribution is at least 1 − α, at the declared coverage level 1 − ε.

GUARANTEE

For any calibration set of size n drawn exchangeably from T, the coverage is at least **1 − ε**, with deviation bounded by 1/(n+1). The bound is exact, distribution-free, and finite-sample.

INPUTS

One calibration set with ground-truth labels (size n, typically 200-500). The deployment set is run black-box; no labels required.

OUTPUTS

The reliability level α, the conformal threshold τ, the abstention rate β. All three are stored in the agent passport.

05 THE METHOD, IN FIVE STEPS

Sample. Score. Calibrate. Threshold. Deploy.

A practitioner can implement the method from this page. The open-source reference implementation handles the bookkeeping and provides the diagnostics, but the algorithm is small. The footnote on each step gives the parameter ranges Cohorte uses in enterprise installations.

STEP 01
Sample
Query the system k times per input. Record the answer set, with frequencies.

k = 5 to 32. Sequential stopping reduces k adaptively.

STEP 02
Score
Compute the non-conformity score s. For each calibration point, s = 1 – frequency of the correct answer.

No logits required.

STEP 03
Calibrate
Find the threshold τ at quantile 1 – ε of the s-distribution on the calibration set.

n = 200 to 500.

STEP 04
Threshold
At deployment, return the answer set above the frequency floor (1 – τ). Empty set means abstain.

Coverage ≥ 1 – ε.

STEP 05
Deploy
The reliability level α is reported. The abstention rate β is reported. Both are gated by the registry.

Reproducible. Auditable.

Why this works without ground truth at deployment. Calibration uses ground truth on the calibration set only. At deployment, the system runs black-box: no labels are required for the live inputs. The guarantee transfers because the calibration set was drawn exchangeably from the same distribution. The exchangeability assumption is the one assumption the method makes, and it is testable when the input distribution can be characterised.

The full proof structure, the case where exchangeability is violated, the partial-coverage extension for hard tasks, the early-stopping algorithm that recovers ~50% of API costs without losing the guarantee, and the open-source implementation are documented in the source paper (Mouzouni, 2026, preprint) and in the **TrustGate** reference repository at **github.com/Cohorte-ai/trustgate**. The architectural point on this page is that the method is small, the dependencies are minimal, and the integration into an existing system is measured in days, not quarters.

06 MEASURED RESULTS

What the bar looks like on five benchmarks.

The reliability level reads as a percentage. The numbers below are from the source paper, computed on standard public benchmarks across three model families (a snapshot of the models tested; the method itself is model-agnostic). They are intended as anchors, not as targets. A given enterprise system will earn its own bar against its own evaluation set; the public-benchmark numbers calibrate the reader's expectations of the method.

SYSTEM	TASK	RELIABILITY LEVEL	COVERAGE ON SOLVABLE
GPT-4.1	GSM8K	**94.6%**	≥ 0.93
GPT-4.1	TruthfulQA	**96.8%**	≥ 0.93
GPT-4.1-nano	GSM8K	**89.8%**	≥ 0.93
GPT-4.1-nano	MMLU	**66.5%**	≥ 0.93
Claude (mid-tier)	Mixed bench	**92%**	≥ 0.93

> Weaker models earn lower reliability levels. The method does not flatter the model. It makes the model's actual quality visible.

The 66.5% number is informative. It says that GPT-4.1-nano on MMLU is not a system that should be deployed at the 95% bar. The naïve accuracy figure on the same benchmark would be in the high 70s. The reliability level surfaces the gap. A deployment decision that treated the accuracy figure as the reliability would systematically over-trust the system. The method makes that over-trust visible *before* the failure shows up at the gate.

The same logic applies in the other direction. A frontier model on a workflow-specific task often earns a reliability level in the high 90s. The number is not a vanity score; it is the evidence the deployment committee needs to authorise the system to operate without human review on the routine cases, while the cases that fall below the threshold are routed to the review queue.

07 BUILDING THE EVALUATION SET

The benchmark you ship against is yours, not public.

Public benchmarks calibrate the method. They do not calibrate the system. A workflow-specific evaluation set is the load-bearing artefact for a defensible deployment. The set has to be drawn from the workflow's actual distribution, labelled with the right answers, and sized for the bar the system is asked to meet.

Source the inputs from the workflow. A summarisation system for credit memos draws its set from credit memos. A scoping assistant for security incidents draws from anonymised incident transcripts. The temptation to start from "similar" public data is the temptation to evaluate a different system than the one being deployed.

Label what the right answer would be. A senior practitioner sits with the calibration set and writes the answer the system should have produced. This is the slow part. It is also the irreplaceable part. The reliability level the system earns is a guarantee against this set of answers; if the answers are wrong, the guarantee is misleading.

Size for the bar. A 95% reliability level at 95% coverage requires a calibration set of approximately 200 labelled points. Higher bars require larger sets. Sub-domains within the workflow may require separate sets if the system behaves differently on them.

Version the set. The set evolves as the workflow evolves. A versioned calibration set with provenance, labelling rules, and a changelog is the audit artefact. A spreadsheet someone last touched six months ago is not.

CALIBRATION SET CHECKLIST

Provenance. Where each input came from. The right to use it. The PII handling.

Labelling rules. Written. Versioned. The rules that resolve disagreements between labellers.

Size. n ≥ 200 for 95% bar. n ≥ 500 for 99% bar.

Sub-domains. If the workflow has sub-types with different difficulty, consider separate sets per sub-type.

Cadence of refresh. Weekly for high-risk systems. Monthly for moderate. Quarterly for low.

Exchangeability check. Sample a fresh batch from production occasionally. If the distribution has shifted, the calibration set is stale.

> **The labelling step is where the practice lives.** The single most discriminating signal between teams that operate trustworthy AI and teams that produce trustworthy slide decks is whether the senior practitioner actually labelled the calibration set, or whether the set was assembled by someone whose judgement is not the system's stake-holder. The labels embody the workflow's standard of "right". They cannot be delegated to a vendor.

08 ONLINE EVALUATION

Shadow mode. Canary. A/B. Roll-out.

Offline evaluation produces the bar. Online evaluation produces the evidence that the bar holds under production conditions. The four staged modes belong to the standard MLOps playbook; what changes for a trustworthy AI system is the metric the stages are watching. They are watching the reliability level, the abstention rate, the drift signal, and the incident log, not just the loss curve.

STAGE	WHAT IT ANSWERS	WHAT GATES IT
Shadow mode	Does the system behave on live data the way it behaved on the calibration set?	Reliability level on the shadow set matches the calibration estimate within the declared slack. Abstention rate is in the expected range.
Canary	Does the system survive a small slice of production traffic with named users on the other side?	A small named cohort runs end-to-end. The incident log stays empty for a defined window. The mind-in-the-loop review confirms the surfaced cases are the right ones.
A/B	Does the system outperform the prior baseline on the workflow's downstream metric?	A pre-registered hypothesis on a downstream metric, an analysis plan, and a stop rule. No quitting halfway through because the early numbers look good or bad.
Roll-out	Does the system hold its bar at full production scale, across the diversity of inputs the calibration set may have under-sampled?	Continuous monitoring kicks in. The registry holds the current bar. Re-calibration cadence is locked. The roll-out completes when the monitoring evidence over the declared window confirms the bar.

The point of staging is not slowness. The point is to make each transition refusable. A system that fails shadow does not enter canary. A system that fails canary does not enter A/B. A system that fails A/B does not enter roll-out. Each gate has its own evidence, its own decision-maker, and its own retreat path. The teams that operate this discipline ship faster on the net because they ship fewer reversals.

09 THE DEPLOYMENT GATE

The bar declared once. Re-tested forever.

The deployment gate is where verification stops being a project and becomes an operation. The bar is declared in the scoping brief, met (or not) on the calibration set, and re-tested on cadence. The decision the gate makes is binary and the decision is auditable. Either the current reliability level meets the declared bar, or it does not. If it does not, the system is held.

Who declares the bar. The named owner of the system, with sign-off from the executive sponsor and, in regulated environments, the second-line risk function. The bar is not negotiated by the team that built the system in isolation. It is negotiated in the open, with the people whose accountability is downstream of the decision.

Why a binary gate. A continuous metric invites continuous justification. A binary gate forces the conversation upstream. If the bar is not met, the conversation is about whether to renegotiate the bar, retire the system, or invest in the system to meet the bar. The gate's binary makes the conversation honest. The gate's transparency makes the bar a public decision rather than a private opinion.

The hold action. A held system continues to receive its input traffic but its outputs are routed to the review queue rather than to downstream consumers. The traffic continues so the reviewer pipeline does not starve and the team can compare the new model's behaviour to the old. The downstream consumer is shielded until the bar is restored.

WHAT THE REVIEWER SEES

DEPLOYMENT GATE INVOICE-FOLLOWUP V1.4

HELD

reliability 93.1% · bar 95%

Calibration set	**v7 · 500 items**
Last re-test	**today 06:00**
Abstention rate	**4.2%**
Incidents (30d)	**0**

Action: outputs routed to the review queue, owner notified, downstream consumer shielded until the bar is restored.

The same record the audit committee, the regulator and the post-incident reviewer read. Generated automatically, stored in the registry.

10 CONTINUOUS CALIBRATION

The bar holds only as long as the world does not move.

A reliability level computed once decays. The input distribution drifts, the vendor swaps the model, the prompt gets edited upstream by a well-meaning team, a regulatory change shifts the labelling rule for what counts as a correct answer. Calibration is therefore a cadence, not a milestone. The team that runs continuous calibration discovers drift before the regulator does.

RISK TIER	RE-CALIBRATION CADENCE	CALIBRATION-SET REFRESH
High-risk (AI Act Annex III)	**Weekly** on the existing set. Monthly on a freshly sampled set.	25% of points refreshed per month from production samples.
Moderate-risk	**Monthly** on the existing set. Quarterly on a refreshed set.	25% refreshed per quarter. Full refresh annually.
Low-risk	**Quarterly** on the existing set. Annual full refresh.	Annual full refresh, with drift sampling between.
Event-triggered	**Immediate**: model swap, prompt edit, regulatory change, incident.	Sampling at the trigger. Re-calibration before re-deployment.

> A reliability level dated nine months ago is opinion. A reliability level dated last week is evidence.

The cadence is not the same as the audit cadence. The audit may happen quarterly. The calibration that supports it runs weekly. The team that conflates the two cadences ends up with quarterly evidence that the quarterly audit will treat as adequate, while the day-to-day operation runs against a stale bar. The right discipline is to run the calibration on the cadence the system requires and let the audit pull a snapshot whenever it needs one.

11 WHERE EVALUATION FAILS IN PRODUCTION

Five failure modes that the bar alone will not catch.

A reliability level is necessary, not sufficient. The five failure modes below are observed in the field. Each requires a specific countermeasure on top of the bar. A team that has all five countermeasures in place runs an evaluation discipline that holds up to adversarial review.

FAILURE	WHAT IT LOOKS LIKE	COUNTERMEASURE
Label rot	The "right answers" in the calibration set were correct when written. The workflow's standard of right has shifted since.	Calibration-set governance: who can re-label, when, and on what evidence. Versioning. Diff review on every change.
Distribution leak	The calibration set was drawn from training data the model has memorised. The bar is inflated by familiarity.	Provenance check on every calibration point. Hold-out samples drawn fresh from production for cross-validation.
Sub-domain mask	The reliability level is high on average and low on a sub-population the average hides. The failures concentrate where the cost is highest.	Stratified evaluation by sub-domain. Per-stratum reliability levels. The gate fires on the worst stratum, not the mean.
Prompt-edit drift	An upstream team edits the system prompt to fix a bug. The bar from the previous prompt does not transfer.	Treat the system prompt as a versioned artefact. Prompt edits trigger event-based re-calibration. The registry tracks the prompt hash.
Coverage gaming	The system meets its bar by raising its abstention rate. The reliability looks great. The throughput collapses.	Joint monitoring of reliability and abstention. The two together are the truthful picture; either one alone is misleading.

12 WHAT THIS IS NOT

Three claims this briefing refuses to make.

The reliability-level method is a real advance over the four evaluation families that preceded it. It is not a universal solvent. The boundaries below are what the method is honest about. A briefing that did not name them would be selling more than the method delivers.

Not a guarantee on individual outputs. The reliability level is a statement about the distribution of system behaviour, not about any single output. A system at a 95% bar will still be wrong on roughly 5% of inputs that would have been correctly answered at a 100% bar. The deployment decision authorises a class of behaviour, not a stream of certified answers. The gate at the output of the system, the audit trail, and the human review of the cases the system abstains on remain necessary.

Not a substitute for domain testing. The reliability level is computed against the calibration set. The calibration set is a sample of the workflow's distribution. There will be parts of the distribution the sample under-represents, and parts of the distribution the workflow has not yet encountered. Domain testing (red-teaming, adversarial sampling, edge-case generation) remains a separate discipline. The reliability level is the floor on which domain testing builds, not a replacement for it.

Not a security guarantee. The reliability level measures the system's correctness on its declared task. It says nothing about the system's behaviour under adversarial input, prompt injection, tool-misuse, or goal-reframing attacks. Those are the territory of Briefing 03 (Agent Governance in Production) and the exploitation-surface research. A team that ships against the reliability-level bar without the security countermeasures has built half the architecture.

> A method that knows its boundaries is the only kind worth deploying inside a regulated environment.

13 HOW THIS LANDS IN A COHORTE ENGAGEMENT

Verification installed. By week seven.

In the Team Bootcamp, the reliability-level method is taught in weeks five to seven and installed on the participant's own system by week eight. In the AI Readiness Program, the same content is installed across the organisation's portfolio. The cadence below is from the standard 12-week Team Bootcamp.

WEEK	WHAT GETS TAUGHT	WHAT GETS SHIPPED
Week 5	Why benchmarks fail. The four families of evaluation. The asymmetry of honesty.	A two-page evaluation strategy for the participant's chosen system, signed by the executive sponsor.
Week 6	Conformal prediction at the practitioner level. The construction of the reliability level.	A labelled calibration set (n ≥ 100 by the end of the week, n ≥ 200 by week eight).
Week 7	The TrustGate library. Self-consistency sampling. Sequential stopping. The deployment report format.	A first reliability level on the participant's system. The bar declared. The gap to the bar named.
Week 8	Shadow-mode discipline. The calibration-set refresh cadence. Sub-domain stratification.	The system in shadow mode with the bar wired in. The first re-calibration scheduled.
Week 9	Stress-testing patterns. The five failure modes named earlier. Countermeasures.	A stress-test report. The countermeasures installed. The system promoted to canary if the report supports it.

The capstone produces evidence, not slides. A graduating team leaves the bootcamp with a working system, a current reliability level, a calibration set with version history, a deployment report, and a calibration cadence on the calendar. The capstone defence is the deployment report. The committee asks for it. The graduate hands it over.

14 FROM THE FIELD

PwC France & Maghreb. The method, installed.

The reliability-level method is the verification primitive behind sixty-plus production systems in the PwC France & Maghreb AI Factory. The architect of those systems is the same person who wrote the source paper. The named reference (Patrick Monteiro, CIO) will take a call arranged after a mutual NDA.

SCOPE	60+ AI systems shipped to production across audit, advisory, consulting, and transaction services. **Verification installed as a layer, not as a project.**
ADOPTION	4,000+ Microsoft Copilot users across the partnership. **+80% adoption increase over six months.** The lift is the consequence of trustworthy systems, not the marketing of them.
GOVERNANCE EVIDENCE	Each system carries an agent passport, a reliability level, and a calibration cadence. The conformity-reporting baseline was prepared **before** the EU AI Act enforcement date.
REFERENCE CALL	Patrick Monteiro, CIO PwC France & Maghreb. Arranged after a **mutual NDA**. The customer takes the call directly.

> The credible verification practice is the one that survives the regulator's first audit. The PwC installation has.

15 REFERENCES

References. The method, sourced.

The papers, repositories and frameworks behind the claims on the previous pages. The full record lives at **teams.cohorte.co/research**.

Mouzouni, C. (2026). Black-Box Reliability Certification for AI Agents via Self-Consistency Sampling and Conformal Calibration. Preprint (2026). The source paper for this briefing. Open-source implementation at github.com/Cohorte-ai/trustgate.

Vovk, V., Gammerman, A., & Shafer, G. (2005). Algorithmic Learning in a Random World. Springer. The foundational text on conformal prediction. Chapter 2 contains the distribution-free guarantee underpinning the reliability-level method.

Angelopoulos, A. N., & Bates, S. (2023). Conformal Prediction: A Gentle Introduction. *Foundations and Trends in Machine Learning, 16*(4). The most accessible monograph for practitioners.

Wang, X. et al. (2023). Self-Consistency Improves Chain-of-Thought Reasoning in Language Models. *ICLR 2023*. The original self-consistency paper, used here as a sampling primitive rather than as the deployment gate.

Zheng, L. et al. (2023). Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *NeurIPS 2023*. The reference characterising LLM-judge bias (position, verbosity, self-preference).

Liu, N. F. et al. (2023). Lost in the Middle: How Language Models Use Long Contexts. *TACL*. The U-shaped performance curve cited in robustness discussions.

Hendrycks, D. et al. (2021). Measuring Massive Multitask Language Understanding. The MMLU benchmark used as anchor in the results table.

Cobbe, K. et al. (2021). Training Verifiers to Solve Math Word Problems. The GSM8K benchmark used as anchor.

European Union (2024). Regulation 2024/1689 on AI. Article 9 (risk management), Article 15 (accuracy, robustness, cybersecurity). The conformity-reporting baseline this briefing supports.

ISO/IEC (2023). ISO/IEC 42001 clauses 8.2-8.5. AI system performance, evaluation, and continuous improvement.

NIST AI RMF (2023). Measure function. The taxonomy of measurement activities mapped onto the practitioner workflow.

The AI OS newsletter. Letters 71, 72, 76 (2026). The asymmetry of honesty; the verification skills gap; the cost of plausible-but-wrong. Archive at charafeddine.co/letters.

FOR THE PERSON DEFENDING THE BAR

Want to walk a system through TrustGate before the briefing ships?

Sixty-minute working session. Bring the system, the workflow context, and the binding constraint. We compute a reliability level, name the gap to the bar, and you leave with the deployment-report skeleton.

charafeddine@cohorte.co

Cohorte SAS · Société par actions simplifiée, registered in France · founded September 2022 · Paris & Rabat

Source paper, open-source implementation, calibration notebooks · all at **teams.cohorte.co/research**