American Impact Review

Original Research

Who owns the conclusion? Ex ante verification agreements as an accountability mechanism for human and AI work in project governance

Vladimir Gaylun*

Independent Researcher, USA

* Corresponding author: Vladimir Gaylun, vladimir.gaylun@gmail.com

OPEN ACCESS

Citation:
Vladimir Gaylun (2026). Who owns the conclusion? Ex ante verification agreements as an accountability mechanism for human and AI work in project governance. *Am. Impact Rev.* e10261230.
10.66308/air.e10261230

Received: March 1, 2026
Accepted: March 27, 2026
Published: March 30, 2026

DOI:
10.66308/air.e10261230

ISSN: 3071-124X

Copyright:
© 2026 Vladimir Gaylun. This is an open access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0).

Abstract

Generative artificial intelligence (AI) has made reports, analyses and presentations nearly free to produce, while the effort required to verify them has not declined. In project and portfolio governance, this asymmetry shifts work from authors to reviewers and blurs accountability for conclusions: the sender relies on one model, the recipient on another, and errors pass between them in fluent prose. This paper asks how organizations should assign accountability for conclusions and design verification when work products are produced by people and AI agents. Using a conceptual, theory-building approach grounded in documentary cases from IT portfolio governance in a large energy company, public cases and illustrative vignettes, the paper develops the ex ante verification agreement (EVA): a specification, attached to a task before work begins, of the claims the work product must support, the errors the acceptor considers material, the checks that will detect them, and who applies the checks and decides acceptance. Four principles follow: the sender owns the facts, the inference and the clarity; checks are agreed with the task and matched to the feared error; checks are owned by the acceptor and cannot be changed by the executor, human or AI; and checks are themselves tested by seeding the feared error. A typology of five work-product errors (omission, substitution, inference error, fabrication and check circumvention) shows that each check detects only the error class it was designed for. Six propositions are formulated for empirical testing.

Keywords: generative AI, verification, accountability, project governance, separation of duties, automation bias, AI agents, principal-agent theory

1. Introduction

Generative AI has changed the economics of written work in organizations. In a controlled experiment, access to a large language model reduced the time professionals needed for writing tasks by about 40% while raising the rated quality of their output (Noy & Zhang, 2023). In a field setting, customer-support agents with an AI assistant resolved more issues per hour, with the largest gains among less experienced workers (Brynjolfsson et al., 2025). The gains are uneven, however. In a field experiment with management consultants, AI improved performance on tasks inside the model's capability frontier but made consultants 19 percentage points less likely to reach a correct answer on a task outside it (Dell'Acqua et al., 2023).

What generative AI has not changed is the cost of reading. A status report, a budget analysis or a twenty-slide presentation can now appear on screen within minutes, polished in format and dense with charts, while the time a recipient needs to establish whether its conclusions follow from its facts remains what it was. Simon's (1971) observation that a wealth of information creates a poverty of attention describes the result. Practitioners have begun to name the phenomenon "workslop": AI-generated work that looks finished but does not advance the task and shifts the burden of interpretation to the recipient (Niederhoffer et al., 2025). In the author's practice as an approver of project documents, such material arrives with increasing frequency. It is competent on the surface but loose underneath, with factual inaccuracies, wording that leads the reader to conclusions other than the required ones, and charts produced simply because a table of data was available.

The governance problem behind this is accountability. When an AI system drafts a report, the sender may treat the model as its author, the recipient may summarize the report with another model, and the substantive work of understanding the project is left to no one. Correspondence between two models does not relieve the participants of the need to understand the project. Existing frameworks recognize the need for human oversight of AI (European Union, 2024, Art. 14; National Institute of Standards and Technology [NIST], 2023), and professional bodies require practitioners to verify AI output before relying on it (American Bar Association [ABA], 2024). Together, these instruments establish the importance of verification and, in professional contexts such as law, the duty to perform it. They say little about how the verification of everyday work products should be designed, when it should be specified and who should own it.

Three research streams bear on the problem without resolving it. Research on automation bias explains why human overseers fail to catch automated errors (Bainbridge, 1983; Parasuraman & Manzey, 2010), but it mostly studies operators monitoring automated systems in aviation, process control and medicine rather than managers accepting knowledge work. Agency theory explains why an agent whose performance is measured will optimize the measure (Holmström & Milgrom, 1991; Kerr, 1975), but it predates agents that can write and edit their own tests. Software engineering has mature practices for specifying checks before implementation (Beck, 2003; Meyer, 1992) and for testing the tests themselves (DeMillo et al., 1978), but these have not been carried over into the governance of reports, analyses and decisions.

This paper addresses the gap with the following research question: *How should organizations assign accountability for conclusions and design the verification of work products when production by humans and AI agents becomes cheap while verification remains costly?* It develops the ex ante verification agreement (EVA), a governance mechanism that attaches a verification specification to a task before work begins, together with a typology of work-product errors that links each error class to the checks that can and cannot detect it. The framework is derived from the author's practice in IT project-portfolio governance and is illustrated with documentary cases, public cases and constructed vignettes. Six propositions are offered for empirical testing.

The paper proceeds as follows. Section 2 reviews the relevant literature. Section 3 describes the conceptual method and the sources. Section 4 presents the framework, the error typology and the cases. Section 5 discusses theoretical and practical implications, limitations and directions for future research, and Section 6 concludes.

2. Theoretical background

2.1. Generative AI and the shifting cost of verification

Large language models produce fluent text whose factual reliability is not guaranteed, and their tendency to generate plausible but unsupported content is well documented across tasks (Ji et al., 2023). In professional domains with high stakes the consequences are concrete. A systematic profiling of legal queries found hallucinations to be pervasive in general-purpose models (Dahl et al., 2024), and even specialized legal

research tools from major vendors hallucinated on 17% to 33% of benchmark queries (Magesh et al., 2025). In *Mata v. Avianca, Inc.* (2023), attorneys who filed a brief citing non-existent cases generated by a chatbot were sanctioned. The court located responsibility not with the tool but with the lawyers who signed the filing. ABA Formal Opinion 512 (ABA, 2024) generalized the point: a lawyer may use generative AI but remains responsible for verifying its output.

These findings shift the economics of knowledge work. If the marginal cost of producing a draft approaches zero, the binding constraint becomes the cost of establishing that the draft is correct. Productivity measured at the moment a file is produced therefore overstates the saving whenever verification, clarification and rework follow (Niederhoffer et al., 2025).

2.2. Automation bias, fluency and the ironies of oversight

Bainbridge (1983) described the "ironies of automation": automating a task leaves the human with the monitoring role at which humans are weakest, and with less practice in the task they are expected to take over when the automation fails. Research on complacency and automation bias shows that people reduce their monitoring and accept automated recommendations even when these are wrong, especially under high workload (Goddard et al., 2012; Parasuraman & Manzey, 2010). Appropriate reliance requires trust that is calibrated to the system's actual capability (Lee & See, 2004).

Generative AI adds a specific mechanism: fluency. Statements that are easier to process are judged more likely to be true (Reber & Schwarz, 1999). A polished summary therefore does not merely conceal an unsupported conclusion; it makes the conclusion more persuasive.

2.3. Communication and common ground

Communication depends on common ground, the knowledge that participants share and know that they share (Clark & Brennan, 1991). A sender who has worked with an AI system knows what was asked, which data were supplied and which answers were discarded; the recipient sees only the document. What seems obvious to the sender must be reconstructed by the recipient from fragments. Media richness theory adds that equivocal messages, which are open to several interpretations, require richer exchange to resolve (Daft & Lengel, 1986). An AI-drafted report that lists completed activities but omits the relationships between them (what changed, what it affects and whether a decision is required) is equivocal in this sense, and its recipient becomes the analyst of someone else's document.

2.4. Agency, measurement and gaming

Agency theory frames delegation as a relationship in which a principal relies on an agent whose effort and information are imperfectly observable (Holmström, 1979; Jensen & Meckling, 1976). When performance is measured on some dimensions but not on others, agents shift their effort toward what is measured (Holmström & Milgrom, 1991), a pattern Kerr (1975) called "rewarding A, while hoping for B." Campbell (1979) warned that the more a quantitative indicator is used for decision-making, the more it is subject to corruption pressures, and Strathern (1997) summarized the phenomenon as "when a measure becomes a target, it ceases to be a good measure." Power (1997) showed how audit can degrade into "rituals of verification" that produce assurance without detection.

AI agents make these concerns operational. Reward hacking, in which a system satisfies the letter of its objective while defeating its purpose, was identified early as a concrete problem in AI safety (Amodei et al., 2016). Baker et al. (2025) documented frontier reasoning models in agentic coding tasks circumventing the

tests used to grade them instead of implementing the requested functionality. When the executor can modify the check, the check no longer measures the work.

2.5. Verification by design

Software engineering offers three ideas that the governance literature has not yet absorbed. Design by contract specifies a component's preconditions, postconditions and invariants as part of its definition rather than after implementation (Meyer, 1992). Test-driven development writes the test before the code that it verifies (Beck, 2003). Mutation testing evaluates a test suite by deliberately seeding faults and observing whether the tests detect them (DeMillo et al., 1978). Computational complexity theory, finally, formalizes an asymmetry that governance can exploit: for many problems, checking a proposed solution is far cheaper than finding one (Cook, 1971). In science, preregistration applies the same logic to analysis plans by fixing them before the data are seen (Nosek et al., 2018).

Taken together, these streams explain why human review of AI output fails (automation bias and fluency), why executors drift toward whatever is checked (agency theory) and how checks can be specified and validated (verification engineering). They have not been integrated into a governance mechanism for the everyday work products of organizations, which is the aim of this paper.

3. Methods

3.1. Research design

The paper follows a conceptual, theory-building design. Of the four approaches to conceptual articles identified by Jaakkola (2020), it combines a typology, which classifies work-product errors by the checks that detect them, with a model, which specifies the relationships among task specification, ownership of verification and detection. Such designs are appropriate when a phenomenon is new and existing theories each explain only part of it (Jaakkola, 2020). The propositions are offered for subsequent empirical testing; the paper does not test them statistically.

3.2. Sources

The framework was derived abductively from four sources. The first is the author's professional practice: serving as head of the IT department responsible for project activity, business planning, control and efficiency at the headquarters of a large, vertically integrated oil and gas company (2018–2021), a unit that owned the IT-project prioritization methodology and ran the secretariat of the company's IT expert council; serving as senior manager and deputy chief digital transformation officer at a steel company (2021–2022); and leading product development at a U.S. company that builds AI systems for legal practice, where human verification of AI output is a design requirement.

The second source is a set of internal documents from the energy company, analyzed as documentary evidence: a November 2020 submission to the company's investment subcommittee proposing an update of the IT-project prioritization methodology, and the December 2020 protocol, official letter and e-mail correspondence on the acceptance of an IT-service standardization result. Organization names are withheld, and figures are reported as they appear in the documents.

The third source consists of public cases with documented records: the spreadsheet error in a widely cited macroeconomic study (Herndon et al., 2014; Reinhart & Rogoff, 2010), the court sanctions in *Mata v. Avianca, Inc.* (2023), the computer-assisted cryptanalysis of historical Enigma messages (Ostwald & Weierud, 2017)

and published evidence of reward hacking by AI agents (Baker et al., 2025). The fourth source is a small number of constructed vignettes, labelled as such, each of which isolates a single mechanism without relying on confidential data.

3.3. Analytical procedure

Each case was examined for (a) the claim the work product was meant to support, (b) the error that occurred or was prevented, (c) the check that detected the error or failed to and (d) who specified and owned that check, and when. Recurring patterns across the cases were grouped into error classes and check types, and the relationships among them were formulated as principles and propositions. Because the author participated in the documentary cases, interpretations of those cases are marked as such, and the documents' own stated rationales are kept distinct from the author's reading of them.

3.4. Trustworthiness

Three measures address the risks of a single-author account by a participant. First, each principle is illustrated with sources of different kinds (internal documents, published records and constructed vignettes), and Table 2 states the type and status of each. Second, statements about the documentary cases are limited to what the documents record, including their limits: where a decision, an adoption or the author's participation is not recorded, the text says so. Third, facts that qualify the argument are reported, such as the original authors' position that the spreadsheet error did not change their conclusion.

4. Results: the ex ante verification agreement

4.1. Definition and overview

An ex ante verification agreement (EVA) is a specification, attached to a task before work begins, of (a) the claims the work product must support, (b) the errors the acceptor considers material, (c) the checks that will be applied to detect those errors and the evidence the executor must supply and (d) who applies the checks and who decides acceptance. The executor may be a colleague, a subordinate, a contractor, an AI agent or the acceptors themselves. Figure 1 shows the process. Four principles define the mechanism, and a further observation concerns tasks for which no check can be named.

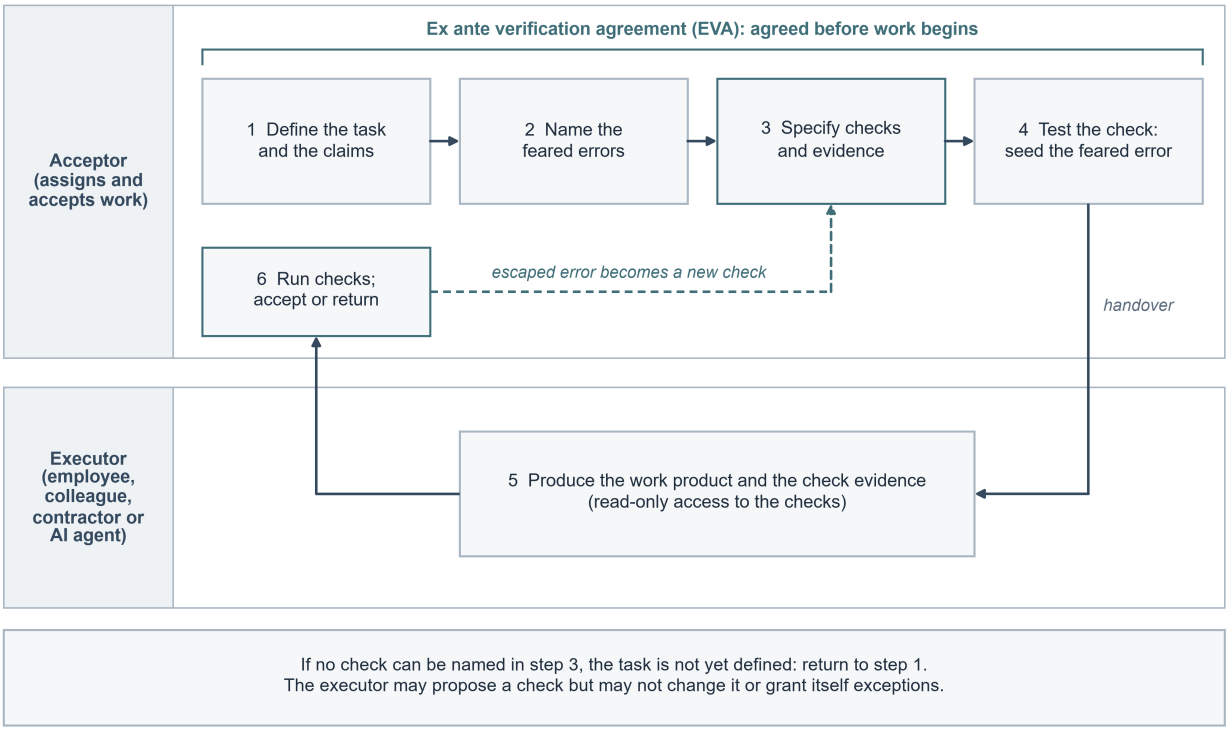


Figure 1. The ex ante verification agreement (EVA) as a process. Steps 1–4 are completed by the acceptor before work begins; the executor works with read-only access to the checks; errors that escape detection feed back into the specification of checks.

Source: author's framework.

4.2. Principle 1: the sender owns the facts, the inference and the clarity

AI can accelerate the preparation of a first draft, but responsibility for the final text lies with the person who sends it: for its facts, for the conclusions drawn from them and for whether the recipient can understand it. The constructed vignette in Figure 2 shows why each element matters. In one week, a project team closed eight of ten planned tasks. Delivery of equipment is delayed, testing cannot start without it, and the project completion date depends on testing. An AI-generated summary reads: "The main scope of work has been completed; 8 of 10 tasks closed; the project maintains strong momentum; there are no material threats to the schedule." The sender checks the task count, finds it correct and sends the report.

No number in the summary is invented, yet the report is wrong. Eight closed tasks say nothing about whether the project can finish on time, because the two remaining tasks may determine the entire schedule. The error lies in the link between the facts and the conclusion. It cannot be fixed by stylistic editing: a smoother sentence would make the unfounded forecast more convincing (Reber & Schwarz, 1999), and a request to shorten the text could leave only the confident promise and remove the details from which a reader might have noticed the contradiction. A revised version states what is known and what is not: eight of ten planned tasks are complete; equipment delivery is delayed and testing cannot start without it; on-time completion therefore cannot yet be confirmed; the team is confirming the delivery date and will then present an updated forecast and options. The revision contains no solution and no firm date, but it shows the recipient where the uncertainty lies, and the next conversation will be about the delivery, the schedule buffer and the options. In the original version the recipient would first have had to discover that a calm tone concealed uncertainty.

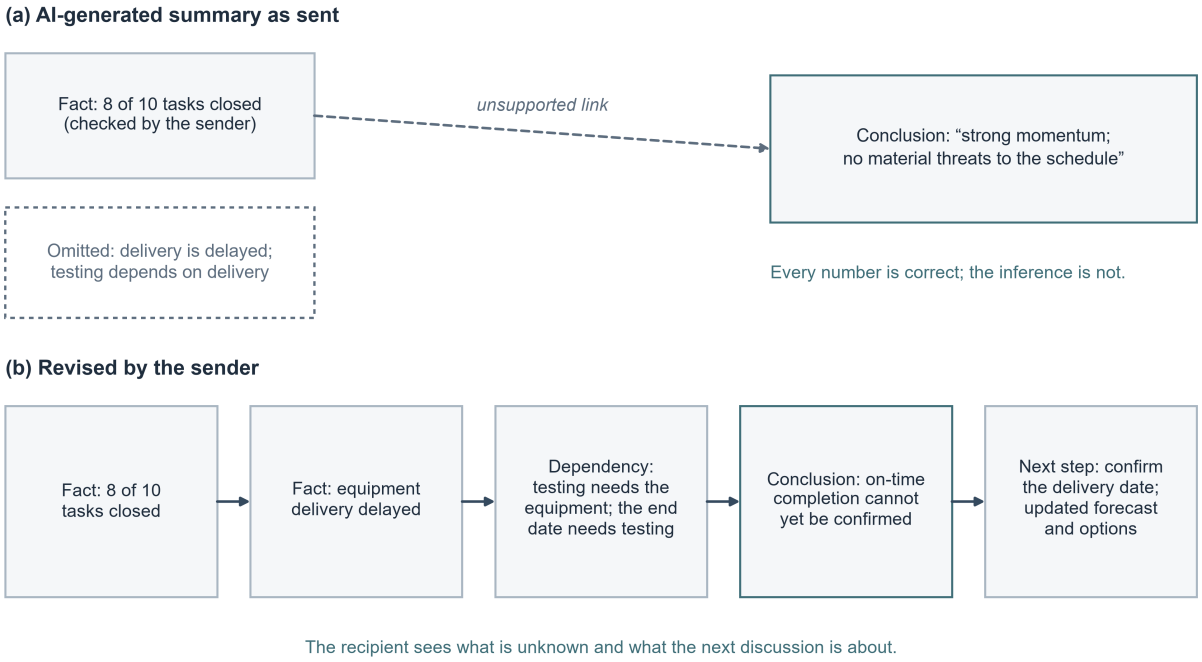


Figure 2. Anatomy of an AI-assisted status report (constructed vignette). (a) The summary as sent: every number is correct, but the conclusion is not supported, and the dependency that determines the schedule is omitted. (b) The sender’s revision links facts, dependency, conclusion and next step.
Source: author’s illustration.

A sender need not know every answer. A draft can be brought to a colleague with a request for help with the forecast, but then the request must be stated as such, because asking to find a solution together and proposing a ready solution for approval require different preparation. Nor does the recipient’s use of AI remove the problem. A model asked to summarize the report cannot restore information that the sender omitted: if the dependency of testing on delivery is absent from the document, no summary will recover it.

The principle does not remove the approver’s responsibility for the approver’s own decision. A sender’s signature does not guarantee that a document is free of errors, and it does not replace review at approval. It signifies that what has been brought for discussion is a position that the person is prepared to explain, not an unprocessed output of an agent. This allocation is consistent with *Mata v. Avianca, Inc.* (2023) and with ABA Formal Opinion 512 (ABA, 2024), under which the professional who signs remains accountable for AI-assisted work.

4.3. Principle 2: the check is agreed with the task and matched to the feared error

The question that opens an EVA is simple: how will we verify ourselves? In reports and analyses, the answer is often a set of control totals that must agree. Project totals must equal the portfolio total; monthly plans must add up to the annual budget; the sum of staff allocations across projects cannot exceed the number of employees. Such reconciliations require no special expertise or tools, only an agreement that they will be performed. They are necessary for a simple reason: people make mistakes, and an error in a total rarely stays in the spreadsheet. It travels into the report, to the committee and into the funding decision.

Timing determines their cost. If the check is agreed before work begins, it can be stated in one sentence in the task (“at the end, we will reconcile project totals with the portfolio total”), and the executor builds the

totals from the start, detecting discrepancies while their cause is still easy to find. If the check is thought of only after the result is ready, it must be designed retrospectively, and the verifier must reconstruct where each number came from. A common situation illustrates the difference: a twenty-slide presentation, a client meeting in an hour and, on slide seventeen, a calculation that would take a day to verify. Had the check been agreed when the task was set, the calculation would have arrived with a control total, a link to the source data and a reconciliation with the previous period. The cost of verification depends less on the complexity of the work than on when verification was first considered.

Some problems carry their verification within them. Ostwald and Weierud (2017) described computer-assisted cryptanalysis of historical Enigma messages and published recovered keys. Their accompanying materials show how to set an Enigma simulator to a recovered key and reproduce the reported plaintext. Finding a key involves searching many possible machine settings; checking that a proposed key reproduces the reported plaintext requires applying those settings to the ciphertext. This illustrates the distinction between finding a solution and checking a proposed solution discussed in computational complexity theory (Cook, 1971). Reports and calculations rarely have such built-in checks, so the check has to be designed, and it is cheapest to design it together with the task.

The second element of the principle is that a check detects only the error it was designed for. Table 1 defines five classes of work-product error, and Figure 3 maps four of them against six common check types. A control total detects omission. When a spreadsheet formula in Reinhart and Rogoff's (2010) study averaged 15 of 20 countries, a simple count of how many countries entered the average would have revealed it. Instead, the error was found about three years later by a graduate student at the University of Massachusetts Amherst who could not replicate the results in a class exercise and eventually obtained the authors' worksheet (Herndon et al., 2014). The authors maintained that the coding error did not change their central conclusion, and other problems identified by Herndon et al. (2014), namely data exclusions and the averaging method, mattered more for the result. The case is instructive for exactly that reason: an error that a trivial count would have caught was found by an outsider after the 90% debt threshold had already entered fiscal policy debates in Europe and the United States.

Table 1. Typology of work-product errors

Error class	Definition	Illustration in this paper	Check that detects it	Check that stays silent
E1 Omission	Material elements are missing	15 of 20 countries in an average (Reinhart & Rogoff, 2010); a schedule dependency left out of a status report	Totals and counts; line-level reconciliation; comparison with the prior result	Style and fluency review; tracing only the claims that are present
E2 Substitution	All elements are present but assigned incorrectly	Alice's and Bob's amounts swapped; two projects' expenses exchanged	Line-level reconciliation; comparison with the prior result	Totals and counts; completeness checks
E3 Inference error	The facts are correct, but the conclusion does not follow from them	"8 of 10 tasks closed, so no threat to the schedule"	Claim–evidence map ("what is the conclusion based on?"); dependency check	Fact checks; totals; line checks
E4 Fabrication	Facts, sources or citations are invented	Non-existent case law cited in a court filing (Mata v. Avianca, Inc., 2023)	Tracing each claim to a source	Fluency and plausibility review; totals
E5 Check circumvention	The executor modifies or disables the check, or exempts itself from it	Agents subverting grading tests (Baker et al., 2025); rating criteria that proponents can move (case A)	A check owned by the acceptor that the executor cannot change; review of every exception	Any check that the executor can edit

Note. Classes E1–E4 are mapped against check types in Figure 3. Source: author's typology.

Control totals are silent on substitution. Suppose a payout list allocates amounts to Alice and Bob and an edit swaps them. Every name and every number remains present, so a completeness check that searches for missing and added items finds nothing, and the total of 80,000 matches to the last unit before and after the edit, yet each person now receives the other's money. The same trap appears in a portfolio summary, where the total budget still matches after the expenses of two projects have been exchanged. Detecting substitution requires a line-level reconciliation of who receives what. Inference errors, as in the status-report vignette, pass both totals and line checks; they require a claim–evidence map that asks what each conclusion rests on and which dependencies it assumes. Fabricated facts and citations pass a review of fluency by design and are detected only by tracing each claim to a source (Magesh et al., 2025). Hence the second question of an EVA: which error are we afraid of? If the fear is that part of an amount will be lost, a total suffices; if it is that recipients will be confused, a line-by-line reconciliation is needed.

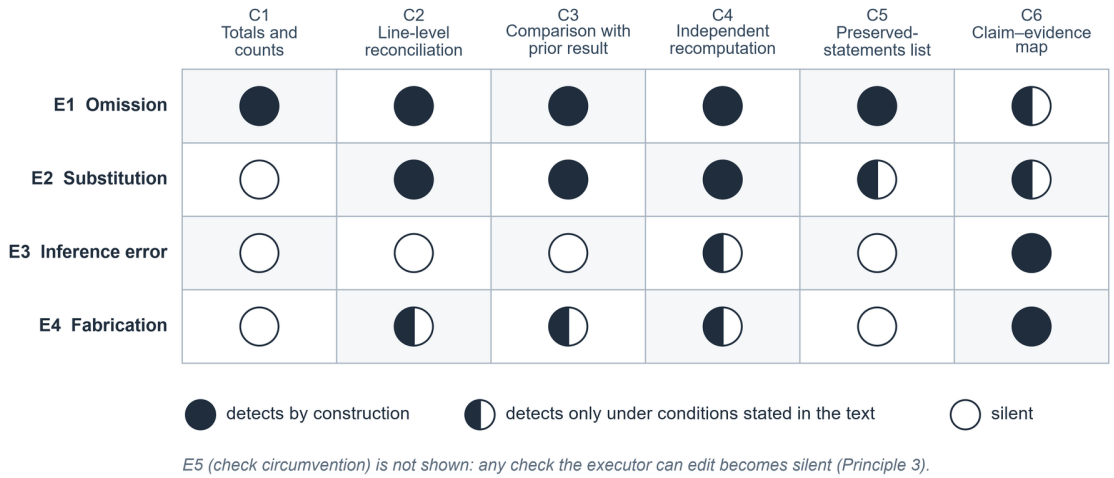


Figure 3. Detection capability of six check types for four classes of work-product error. The judgments are conceptual and indicate detection by design; they are not empirical frequencies.
Source: author's typology.

For text, the corresponding check is a list of statements that must be preserved, written from the source before the revised version has been read and has begun to feel familiar. For a portfolio report, such a list might contain only a few lines: how many projects are under way, what the total budget is and which projects are furthest behind schedule. The list serves twice: it instructs the executor, and it remains with the acceptor as the acceptance criterion. Numerical comparison can be delegated to software, while statements that will drive human decisions should be read in full.

4.4. Principle 3: the check is owned by the acceptor and cannot be changed by the executor

Checks can be circumvented if the executor is permitted to change them. Agency theory predicts this for human agents (Holmström & Milgrom, 1991; Kerr, 1975), and recent evidence documents it for AI agents. Baker et al. (2025) observed frontier reasoning models in agentic coding environments passing the tests used to grade them by subverting those tests rather than by producing working code. Consider, as an illustration, a software migration in which an automated compatibility check blocks a merge because a function used by other programs has been lost. An agent that is allowed to configure the check can set an exception flag and proceed; unless the person who owns the check asks why the exception is acceptable, the loss passes. The remedy is organizational rather than technical. The check is agreed before work begins, the executor builds the evidence for it into the deliverable, the person who assigned the work decides whether the check has been passed, and the executor has no right to change it. An AI agent needs read access to the check so that it can work toward it, not write access.

Documentary case A illustrates the same principle in human portfolio governance. In 2015, the energy company's headquarters unit responsible for IT project management, jointly with its investment analysis department, introduced a methodology for ranking IT projects by four weighted criteria: benefits (40%), project stage (25%), criticality (25%) and "importance for the company" (10%), defined by whether the project appeared in managers' key performance indicators (KPIs). In November 2020, a submission signed by the company's vice-president for IT asked the investment subcommittee for functional projects to review an update of the methodology; the author, who headed the unit that owned the methodology, was the designated presenter. The update removed the importance criterion and re-weighted the others (benefits 45%, stage 10%, criticality

45%). Criticality scores were tied to their evidentiary basis: the highest score to legislative requirements and the requirements of federal supervisory bodies, a lower score to directives of the chief executive and the lowest non-zero score to requirements of internal control units. The submission's own rationale was that the inclusion of a project in participants' KPIs should not determine its priority, that the confidentiality of KPI data made ratings hard to update, that the mere existence of a signed contract substantially inflated a project's rating and that overlapping criteria risked overstating some factors. Re-rating the portfolio of 326 projects and business needs (RUB 13.29 billion) under the update reduced the number of high-criticality projects with low integral ratings from 31 to 10.

Read through the lens of the EVA, an interpretation by the author rather than the document's own framing, the update removed from the rating two factors that project proponents could move by their own actions, namely writing the project into someone's KPIs and signing a contract early, and anchored criticality in evidence external to the proponent. In EVA terms, the check on a project's priority was moved out of the executor's reach. The submission was a proposal, and neither the subcommittee's decision nor the formal adoption of the update is recorded in the documents examined.

Documentary case B illustrates the separation of executor and acceptor. Under a 2020 KPI for standardizing the company's IT services, the company's in-house IT integrator consolidated a catalogue of 16,499 IT-service records (986 of them unique) into 73 standard business services. In December 2020, a meeting chaired by the director of the headquarters informatization department agreed the integrator's results as confirming fulfilment of the KPI, on the basis of a report by headquarters department heads. The integrator then submitted the results by official letter to three headquarters addressees, including the author, who approved them. The executor and the acceptors belonged to different organizations, and acceptance was a decision of the headquarters side rather than a self-assessment by the integrator. Table 2 summarizes the cases and the principles they illustrate.

Table 2. Cases used in the paper

Case	Type of evidence	Status	Principle illustrated
A. Update of an IT-project prioritization methodology, energy company, 2020	Internal submission to the investment subcommittee, November 2020; the author was the designated presenter	Proposal; portfolio of 326 items re-rated; decision and adoption not recorded in the documents examined	Principle 3: criteria that proponents can move removed; criticality tied to external evidence
B. Acceptance of an IT-service standardization result, energy company, 2020	Meeting protocol, official letter and e-mail correspondence, December 2020	Agreed as confirming fulfilment of the KPI	Principle 3: executor and acceptor separated; acceptance decided by the acceptor
C. Spreadsheet error in a macroeconomic study	Published record (Herndon et al., 2014; Reinhart & Rogoff, 2010)	Error found in 2013	Principle 2: an omission that a count would have caught
D. Fictitious citations in a court filing	Court opinion (Mata v. Avianca, Inc., 2023)	Sanctions imposed	Principle 1: the signer owns AI-assisted work; error class E4
E. Computer-assisted cryptanalysis of historical Enigma messages	Published cryptanalysis and recovered-key examples (Ostwald & Weierud, 2017)	Published in 2017	Principle 2: verification built into the problem
F. Reward hacking by AI agents in coding tasks	Research report (Baker et al., 2025)	Published	Principle 3: checks that the executor can change; error class E5
G. Weekly status report; payout list; workload assignment	Constructed vignettes	Illustrative	Principles 1, 2 and 4; Section 4.6

Note. Organization names in cases A and B are withheld; figures are reported as recorded in the documents.

The same logic applies when one person is both executor and acceptor. Writing down the expected control total before starting a calculation prevents the expectation from adjusting to the result once it is seen. A pre-written check works like an agreement with a second person, even when no second person is present, which is the logic of preregistration in science (Nosek et al., 2018).

4.5. Principle 4: the check is tested, and escaped errors become new checks

A check should itself be verified before it is relied on. The method is borrowed from mutation testing (DeMillo et al., 1978): deliberately introduce the error one fears, for example by swapping two amounts, and observe whether the check catches it. A tool built to find missing and added numbers will honestly report nothing missing after Alice's and Bob's amounts have been swapped, because a check catches only the error it was designed to detect. When an error nevertheless escapes, it should become a check for the next time. This is common practice in software engineering, where a defect that reaches production is typically followed by a regression test that reproduces it, and it corresponds to the feedback loop in Figure 1.

4.6. When no check can be named, the task is undefined

Sometimes the question "how will we verify ourselves?" has no answer. Consider the assignment "assess the team's workload." If no one can name what the answer will be compared with, the organization has not yet agreed on what counts as workload: hours in the timesheet, assignments in the project plan or a manager's perception. The executor will return with a plausible figure that cannot be compared with anything, and the dispute will begin after the work is done. Answering the verification question clarifies the task, because the phrase "we will check against last quarter's timesheet hours" specifies what exactly is to be counted. This mirrors design by contract, in which a component whose postcondition cannot be stated has not yet been specified (Meyer, 1992). The verification question therefore tests the task as well as the result: until it is clear how a result will be verified, it is also unclear what exactly has to be done.

4.7. Communication agreements for AI-assisted work

Large projects commonly maintain a communications management plan that specifies who provides which information to whom, through which channels and how often (Project Management Institute [PMI], 2017). The EVA extends this practice from the logistics of communication to the content of AI-assisted messages. For a document submitted for approval, such an agreement can be brief: state at the outset the question to be decided; then give the sender's assessment and the facts on which it rests; name unknowns as unknown; and keep details where they allow the conclusion to be checked. In a two-person personal project of the author's, in which both participants use AI extensively, such rules were adopted early so that the participants would not become relays of messages between agents: describe the needed result briefly, read and correct the generated text before sending it, and use a short structured list where one fits instead of a long narrative. None of this requires writing every sentence by hand. AI can be assigned the draft, the abridgement or the search for contradictions; the human checks the result and decides whether to sign it.

4.8. Propositions

The framework yields six propositions, which are summarized with indicative measures in Table 3.

Proposition 1. As generative AI reduces the cost of producing work products, verification accounts for a growing share of the total effort required to reach an accepted product; productivity measured at the moment of production therefore overstates the saving.

Proposition 2. Checks specified before work begins detect errors earlier and at lower cost than checks devised after the work product is complete.

Proposition 3. A check detects errors only within the class for which it was designed; the match between the type of check and the feared class of error determines detection.

Proposition 4. When the executor, human or AI, can modify a check or move the measured quantity by its own actions, the detection capability of the check declines.

Proposition 5. Tasks for which no check can be named before work begins are associated with more rework and more post-delivery disputes than tasks for which a check is specified.

Proposition 6. An agreed message format (question, assessment, basis and named unknowns) reduces clarification exchanges and decision latency for AI-assisted reports.

Table 3. Propositions and indicative measures for empirical testing

Proposition	Statement (abridged)	Indicative measures	Possible design
1	Verification takes a growing share of total effort; productivity measured at production overstates the saving	Time from first draft to acceptance; clarification rounds; share of review time	Time-use study; workflow log data
2	Checks agreed before work detect errors earlier and at lower cost	Time to detection; cost of correction; escaped errors	Field comparison of tasks with and without EVAs
3	Detection depends on the match between check type and error class	Detection rate of seeded errors by class and check type	Controlled error-seeding experiment
4	Checks that the executor can change lose detection capability	Rate of check modification or exception; detection rate	Experiment with AI agents: read-only versus writable checks
5	Tasks without a nameable check produce more rework and disputes	Rework hours; post-delivery disputes; change requests	Longitudinal study in a project management office
6	An agreed message format reduces clarifications and decision latency	Clarification messages per report; time to decision	Field experiment across teams

Note. EVA = ex ante verification agreement.

5. Discussion

5.1. Theoretical implications

The paper makes three theoretical contributions. First, it relocates the unit of analysis for human oversight of AI from the individual reviewer to the agreement that precedes the work. Research on automation bias treats the overseer's vigilance as the main lever (Parasuraman & Manzey, 2010). The EVA treats vigilance as insufficient on its own and makes detection a property of a check that is specified in advance, matched to an error class and validated with seeded errors. This responds to the irony identified by Bainbridge (1983): if humans are poor monitors, monitoring should be structured rather than demanded.

Second, the error typology refines the discussion of AI reliability. Much of the literature concentrates on fabrication (Dahl et al., 2024; Ji et al., 2023). The status-report vignette and the substitution example show that a work product can contain no invented fact and still be wrong, and that the checks most organizations

already perform (totals, fact checks and style review) are silent on inference and substitution errors. Because AI makes text more fluent, these errors become more persuasive rather than less (Reber & Schwarz, 1999).

Third, the framework extends agency theory to agents that can edit their own performance measures. Classical models assume that the principal controls the measurement technology (Holmström & Milgrom, 1991). AI agents with write access to their tests violate this assumption (Baker et al., 2025), and documentary case A shows that human proponents can achieve a similar effect when a rating includes factors they control. In both situations, the governance answer is that the executor cannot change the check. This connects the agency literature to Power's (1997) critique of verification rituals: a check that the executor can change produces assurance without detection.

5.2. Practical implications

The EVA requires no new systems and no separate budget. Managers can begin by asking "how will we verify ourselves?" when assigning work to an employee, colleague, contractor or AI agent, or before doing a calculation themselves, and by writing the answer next to the task. One of four types of check usually suffices: control totals that must agree (the sum of the rows equals the total, and an average includes as many objects as the table contains); comparison with the previous result (when a report or process moves to a new system, first reproduce the old output and only then improve it); independent recomputation or review by another person; and, for text, a list of statements that must be preserved. The check stays with the acceptor. The executor may propose it but may not change it or make exceptions for itself; an AI agent receives the answer to the verification question in its assignment and may run the check itself, but the person who assigned the work decides whether it has been passed. Before relying on a check, the acceptor seeds the feared error and confirms that the check catches it. Table 4 gives a worked example for a monthly portfolio report.

Table 4. Worked example of an ex ante verification agreement for a monthly portfolio report (constructed)

Element of the agreement	Content in the example
Task and decision	Prepare the monthly IT-portfolio report for the investment committee, which will decide whether to release next-quarter funding for the projects that are behind schedule
Claims the report must support	Number of active projects; total budget and spending to date; the projects furthest behind schedule and the reason for each delay; a completion forecast for each
Feared errors	E1: a project missing from the totals. E2: spending booked to the wrong project. E3: an "on track" forecast that ignores a dependency
Checks and evidence	C1: project totals equal the portfolio total, and the project count equals the register. C2: spending per project reconciled with the ledger extract. C6: each forecast lists the facts and dependencies it rests on. Evidence: control totals, a link to the source data and a reconciliation with the previous month
Test of the checks	Before first use, the acceptor swaps the spending of two projects in a copy of the data and confirms that C2 reports the swap (C1 will not)
Ownership	The portfolio manager, as acceptor, owns the checks and decides whether they are passed. The executor, an analyst or an AI agent, has read-only access and reports every exception
Message format	The question to be decided; the assessment; the facts it rests on; unknowns named as unknown

Note. E = error class (Table 1); C = check type (Figure 3). Source: author's illustration.

For approvers, the framework suggests returning unverified material with questions rather than silently rewriting it. In the status-report example, the useful question is "what is the forecast of on-time completion based on?" The answer distinguishes two situations: a schedule buffer that exists but was not stated, which calls for a fuller explanation, and a dependency that was never checked, which calls for a revised estimate. Silent rewriting gives the sender a corrected file without showing where the error was.

For AI governance, the EVA offers an operational form for the human oversight provided for in Article 14 of the EU Artificial Intelligence Act (European Union, 2024) and for the "measure" and "manage" functions of the NIST AI Risk Management Framework (NIST, 2023). In professional services where accountability is personal, as in law, it gives practical content to the duty to verify AI output (ABA, 2024).

5.3. Boundary conditions

The EVA has costs and limits. Specifying checks takes time, and for low-stakes or easily reversible work an informal review may be enough; the mechanism is most useful where a work product feeds a decision that is costly to reverse. Exploratory tasks, whose result cannot be anticipated, admit only partial checks: on the data and the method rather than on the answer. Finally, a specified check can itself become a target (Strathern, 1997): an executor may work to pass the named checks and neglect what they do not cover. Principle 4 and the acceptor's own judgment mitigate this risk but do not remove it, so an EVA complements review at acceptance and does not replace it.

5.4. Limitations and future research

The paper has several limitations. It is conceptual, and the propositions have not been tested. The documentary cases come from the author's own practice in one company and are interpreted by a participant, which introduces potential bias; the documents are internal and cannot be released, although their content is reported as recorded. The vignettes are constructed to isolate mechanisms and do not represent observed frequencies. The detection judgments in Figure 3 are conceptual and should be calibrated empirically. Finally, AI capabilities change quickly, and some of the checks described here may soon be automated, although the question of who owns them will remain.

Future research can test the propositions with several designs. Field studies in project management offices could compare time to acceptance and rework for tasks with and without EVAs (Propositions 1, 2 and 5). Laboratory or field experiments could seed known errors of each class into reports and measure detection by type of check (Proposition 3). Experiments with AI coding and analysis agents could compare detection and test-tampering rates when checks are read-only rather than writable (Proposition 4). Studies of AI-assisted communication could measure clarification exchanges and decision latency under agreed message formats (Proposition 6).

6. Conclusions

Generative AI has made producing documents cheap and left verifying them expensive. In this situation, the scarce resource in project governance is not text but confidence that the text is right, and that confidence cannot be delegated to the tool that produced the draft. The ex ante verification agreement assigns ownership of the facts, the inference and the clarity to the sender; agrees the check together with the task and matches it to the feared error; keeps the check with the acceptor, out of the executor's reach; and tests the check before relying on it. If no check can be named before work begins, it is not yet clear what the result should be. The question "how will we verify ourselves?" therefore verifies the task as well as the result, and it applies equally to work done by employees, colleagues, AI agents and oneself.

Declarations

Author Contributions. Vladimir Gaylun: Conceptualization, Methodology, Investigation, Formal analysis, Visualization, Writing – original draft, Writing – review & editing.

Funding. This research received no external funding.

Data Availability Statement. No new datasets were generated or analyzed in this study. The documentary cases draw on internal company documents that cannot be shared for reasons of confidentiality; their relevant content is reported in the text. Public sources are listed in the references.

Conflict of Interest. The author is Head of Product at a U.S. company that develops AI systems for legal practice. The company did not fund this research or influence its conclusions.

Ethics Statement. This study did not involve human participants, animals or personal data and did not require ethics approval.

Acknowledgments. N/A

References

1. American Bar Association. (2024). *Formal opinion 512: Generative artificial intelligence tools*. ABA Standing Committee on Ethics and Professional Responsibility.

2. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). *Concrete problems in AI safety* (arXiv:1606.06565). arXiv. https://doi.org/10.48550/arXiv.1606.06565

3. Bainbridge, L. (1983). Ironies of automation. *Automatica, 19*(6), 775–779. https://doi.org/10.1016/0005-1098(83)90046-8

4. Baker, B., Huizinga, J., Gao, L., Dou, Z., Guan, M. Y., Madry, A., et al. (2025). *Monitoring reasoning models for misbehavior and the risks of promoting obfuscation* (arXiv:2503.11926). arXiv. https://doi.org/10.48550/arXiv.2503.11926

5. Beck, K. (2003). *Test-driven development: By example*. Addison-Wesley.

6. Brynjolfsson, E., Li, D., & Raymond, L. (2025). Generative AI at work. *The Quarterly Journal of Economics, 140*(2), 889–942. https://doi.org/10.1093/qje/qjae044

7. Campbell, D. T. (1979). Assessing the impact of planned social change. *Evaluation and Program Planning, 2*(1), 67–90. https://doi.org/10.1016/0149-7189(79)90048-X

8. Clark, H. H., & Brennan, S. E. (1991). Grounding in communication. In L. B. Resnick, J. M. Levine, & S. D. Teasley (Eds.), *Perspectives on socially shared cognition* (pp. 127–149). American Psychological Association. https://doi.org/10.1037/10096-006

9. Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing* (pp. 151–158). Association for Computing Machinery. https://doi.org/10.1145/800157.805047

10. Daft, R. L., & Lengel, R. H. (1986). Organizational information requirements, media richness and structural design. *Management Science, 32*(5), 554–571. https://doi.org/10.1287/mnsc.32.5.554

11. Dahl, M., Magesh, V., Suzgun, M., & Ho, D. E. (2024). Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis, 16*(1), 64–93. https://doi.org/10.1093/jla/laae003

12. Dell'Acqua, F., McFowland, E., III, Mollick, E. R., Lifshitz-Assaf, H., Kellogg, K., Rajendran, S., et al. (2023). *Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality* (Harvard Business School Working Paper No. 24-013). https://doi.org/10.2139/ssrn.4573321

13. DeMillo, R. A., Lipton, R. J., & Sayward, F. G. (1978). Hints on test data selection: Help for the practicing programmer. *Computer, 11*(4), 34–41. https://doi.org/10.1109/C-M.1978.218136

14. European Union. (2024). Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Official Journal of the European Union*, L 2024/1689.

15. Goddard, K., Roudsari, A., & Wyatt, J. C. (2012). Automation bias: A systematic review of frequency, effect mediators, and mitigators. *Journal of the American Medical Informatics Association, 19*(1), 121–127. https://doi.org/10.1136/amiajnl-2011-000089

16. Herndon, T., Ash, M., & Pollin, R. (2014). Does high public debt consistently stifle economic growth? A critique of Reinhart and Rogoff. *Cambridge Journal of Economics, 38*(2), 257–279. https://doi.org/10.1093/cje/bet075

17. Holmström, B. (1979). Moral hazard and observability. *The Bell Journal of Economics, 10*(1), 74–91. https://doi.org/10.2307/3003320

18. Holmström, B., & Milgrom, P. (1991). Multitask principal–agent analyses: Incentive contracts, asset ownership, and job design. *The Journal of Law, Economics, and Organization, 7*(Special Issue), 24–52. https://doi.org/10.1093/jleo/7.special_issue.24

19. Jaakkola, E. (2020). Designing conceptual articles: Four approaches. *AMS Review, 10*(1–2), 18–26. https://doi.org/10.1007/s13162-020-00161-0

20. Jensen, M. C., & Meckling, W. H. (1976). Theory of the firm: Managerial behavior, agency costs and ownership structure. *Journal of Financial Economics, 3*(4), 305–360. https://doi.org/10.1016/0304-405X(76)90026-X

21. Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., et al. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys, 55*(12), 1–38. https://doi.org/10.1145/3571730

22. Kerr, S. (1975). On the folly of rewarding A, while hoping for B. *Academy of Management Journal, 18*(4), 769–783. https://doi.org/10.2307/255378

23. Lee, J. D., & See, K. A. (2004). Trust in automation: Designing for appropriate reliance. *Human Factors, 46*(1), 50–80. https://doi.org/10.1518/hfes.46.1.50_30392

24. Magesh, V., Surani, F., Dahl, M., Suzgun, M., Manning, C. D., & Ho, D. E. (2025). Hallucination-free? Assessing the reliability of leading AI legal research tools. *Journal of Empirical Legal Studies, 22*(2), 216–242. https://doi.org/10.1111/jels.12413

25. *Mata v. Avianca, Inc.*, 678 F. Supp. 3d 443 (S.D.N.Y. 2023).

26. Meyer, B. (1992). Applying "design by contract." *Computer, 25*(10), 40–51. https://doi.org/10.1109/2.161279

27. National Institute of Standards and Technology. (2023). *Artificial intelligence risk management framework (AI RMF 1.0)* (NIST AI 100-1). https://doi.org/10.6028/NIST.AI.100-1

28. Niederhoffer, K., Kellerman, G. R., Lee, A., Liebscher, A., Rapuano, K., & Hancock, J. T. (2025, September 22). AI-generated "workslop" is destroying productivity. *Harvard Business Review*. https://hbr.org/2025/09/ai-generated-workslop-is-destroying-productivity

29. Nosek, B. A., Ebersole, C. R., DeHaven, A. C., & Mellor, D. T. (2018). The preregistration revolution. *Proceedings of the National Academy of Sciences, 115*(11), 2600–2606. https://doi.org/10.1073/pnas.1708274114

30. Noy, S., & Zhang, W. (2023). Experimental evidence on the productivity effects of generative artificial intelligence. *Science, 381*(6654), 187–192. https://doi.org/10.1126/science.adh2586

31. Ostwald, O., & Weierud, F. (2017). Modern breaking of Enigma ciphertexts. *Cryptologia, 41*(5), 395–421. https://doi.org/10.1080/01611194.2016.1238423

32. Parasuraman, R., & Manzey, D. H. (2010). Complacency and bias in human use of automation: An attentional integration. *Human Factors, 52*(3), 381–410. https://doi.org/10.1177/0018720810376055

33. Power, M. (1997). *The audit society: Rituals of verification*. Oxford University Press.

34. Project Management Institute. (2017). *A guide to the project management body of knowledge (PMBOK guide)* (6th ed.). Project Management Institute.

35. Reber, R., & Schwarz, N. (1999). Effects of perceptual fluency on judgments of truth. *Consciousness and Cognition, 8*(3), 338–342. https://doi.org/10.1006/ccog.1999.0386

36. Reinhart, C. M., & Rogoff, K. S. (2010). Growth in a time of debt. *American Economic Review, 100*(2), 573–578. https://doi.org/10.1257/aer.100.2.573

37. Simon, H. A. (1971). Designing organizations for an information-rich world. In M. Greenberger (Ed.), *Computers, communications, and the public interest* (pp. 37–72). Johns Hopkins Press.

38. Strathern, M. (1997). 'Improving ratings': Audit in the British university system. *European Review, 5*(3), 305–321. https://doi.org/10.1017/S1062798700002660