Trust, but Verify: Harnessing the AI Bonanza in Industrial Engineering [1]

AI can take over the most labor-intensive and error-prone work of design and engineering. The firms that build verification first will harvest the gains

Konstantin Lagutin

Abstract

Industrial design and engineering (D&E) is full of labor-intensive, error-prone routine work: line lists, datasheets, quantity take-offs, cable schedules, document checks and sorting of design clashes. Generative and agentic AI can now take over much of it and generate dozens of value-engineering alternatives in minutes. The opportunity is large, but it comes with a condition. Checking AI output has not become cheaper: every option that may end up in steel and concrete still needs an engineer who understands it, tests it and signs for it. Chip designers and software teams hit this "verification barrier" first; industrial D&E is next. Drawing on 134 sources (2023-2026), a ranked map of 25 open issues and first-hand evidence from steel, fertilizer and green-steel programs, this paper argues that the binding constraint on AI-created value is now an organization's capacity to verify. It proposes a verify-first value engine for EPC contractors, design institutes and OEM[2] engineering teams. Agents propose; a six-layer stack of source-traceability, rules, physics, memory, red-team and human checks disposes of unsound options, running the fastest and least costly checks first so that expert time goes only to survivors. Engineers decide at review levels matched to risk, six agent controls keep the agents in bounds, and a staged value register counts only savings that independent cost engineers have verified. Most tools are already proven. The paper assigns owners, maps GenAI and verifiers onto the design process, proposes an architecture, and provides a fault-injection test, a verified-trust scorecard and a 24-month roadmap.

Keywords: *generative AI; agentic AI; value engineering; verification; human-in-the-loop (HITL); industrial BIM; engineering management; automation bias; metals and mining*

[1] How to cite this paper: Lagutin, K. (2026). Trust, but Verify: Harnessing the AI Bonanza in Industrial Engineering; Featured Paper, *PM World Journal*, Vol. XV, Issue XI, November.

[2] OEM stands for Original Equipment Manufacturers.

© 2026 Konstantin Lagutin

1. AI Offers Engineering a Bonanza, Provided Its Output Is Verified

Miners call an unexpectedly rich ore body a *bonanza*. Industrial design and engineering is now facing an opportunity of this kind. A large part of every engineering budget goes into routine work that is labor-intensive, repetitive and prone to errors: transferring data from piping and instrumentation diagrams (P&IDs) into line lists, filling in hundreds of equipment datasheets, counting quantities, routing cables, checking drawings against standards, sorting thousands of clashes, etc. Generative and agentic AI can now do much of this work in minutes rather than weeks. At the front end of a project, the same agents can generate dozens of alternatives for layouts, materials, logistics and energy systems, giving value engineering (VE) a breadth of search it never had.

The value is moving to agents, fast. BCG's 2026 survey of 1,330 senior executives found that agentic AI's share of all AI value rose from 17% in 2025 to 22% in 2026 and is expected to reach 39% by 2030. The same survey exposes the constraint: 42% of companies expect agents to act autonomously by 2030, yet only 5% have the relevant agent controls in place today (Apotheker et al., 2026).

Two industries hit this verification barrier first. In September 2026, an executive of a leading AI software company reported that 89% of the code committed in his firm is now written by an autonomous agent, and warned that "human reviewers can become the constraint that code generation used to be" (Lee, 2026). In a survey published the same month, 26% of software developers and 26% of engineering leaders named the review of AI-generated code as their main delivery constraint (Qodo, 2026). Chip designers learned the lesson a generation earlier: by 2022, design engineers spent about 49% of their time on verification, and projects employed, at peak, roughly as many verification engineers as designers (Foster, 2022).

Industrial D&E is next. In 2019, a major East European integrated steelmaker rebuilt the 2D design of a blast-furnace revamp in 3D and let software hunt for collisions. The model found 3,058 clashes in days; 148 were critical. Finding them was the easy part. Deciding which ones mattered, who owned them and how to fix them took a team of engineers, issue by issue. Now imagine that team receiving not 3,058 clashes but 3,000 AI-generated design alternatives. Without a new way of checking, it would drown.

The argument of this paper can be expressed in one sentence: with AI, generation is cheap, but verification is value. The Russian proverb that American president Reagan made famous says it in three words: *trust, but verify*.

> *The next bottleneck in engineering is not drawing. It is believing.*

© 2026 Konstantin Lagutin

1.1 The Bonanza Lies in Routine Work

Table 1 lists the routine operations where AI can take the most labor out of D&E today, why they are error-prone, and which verification layer (Section 5.2) must check the result. Engineering typically costs a high single-digit to low double-digit share of a project's total installed cost (TIC), so even the routine share is worth a great deal. And these operations are where many errors are born: design-induced rework has long been reported to account for more than 70% of all rework in construction and engineering projects (Love et al., 2008).

For example, a leading global resources company received more than 10,000 change requests a year for the master data of its industrial assets. The process was manual, a six-month backlog built up, and data quality degraded. An AI agent now turns a plain-language description of the change into a checked, ready-to-submit request in 3 steps instead of 68; employees fix errors where they spot them, so data quality improves at the source (Apotheker et al., 2026). Engineering data management in a design office looks very similar.

Table 1. Routine D&E Operations Where AI Can Take Out the Most Labor

Routine operation	Why it is labor-intensive and error-prone	What GenAI agents do	Main verification layers
Line lists, equipment lists, instrument indexes	Thousands of entries manually re-entered between documents	Extract and reconcile lists from P&IDs and the model	L1 consistency rules (e.g., pipe specification vs design pressure)
Equipment datasheets	Hundreds of sheets; values copied from older projects	Draft datasheets from process data and standards	L0 source traceability; L2 operating-condition (flow, pressure) calculations
Material take-offs	Manual counting; duplicates and omissions	Take quantities directly from model objects	L1 quantity rules; unique object IDs
Cable and pipe routing, cable schedules	Repetitive; clashes and undersizing	Route, size and schedule	L1 clash checks; L2 electrical load and cable voltage-drop calculations
Drawing and document checks against standards	Reviewer fatigue; uneven quality	First-pass check with marked-up comments	L1 rules; L5 sample review
Sorting of design clashes	Thousands of clashes, few of them critical	Group, rank and assign clashes	L1; L5 for critical clashes
Engineering master data and change requests	Manual, multistep, backlogged	Draft, check and route change requests	L0 source traceability; L1 rules
Value-engineering options at FEL and FEED	Workshops compare only two or three alternatives	Generate 50-100 options on space, materials, logistics, energy and carbon	All six layers

1.2 Why Now

Why is this opportunity opening now rather than five years ago? **Three forces meet in 2026. First**, AI agents have reached the engineer's desk. Agents are AI systems that plan multistep

© 2026 Konstantin Lagutin

tasks, call software tools such as design packages, simulators and cost databases, and check and iterate within the permissions they are given. Autodesk, for example, previewed agentic workflows across its design and construction clouds in September 2026, stressed that its assistant is "designed to support professional judgment through controls and human supervision", and scheduled fuller autonomy for 2027 (Unite.AI, 2026). **Second**, the engineering workforce is under strain: the United States alone counts some 400,000 open positions in architecture, engineering and construction, and two-thirds of owners report understaffed project teams (Fjeldstad et al., 2024; Mishra, 2025). **Third**, the rules are tightening. The licensed engineer remains in responsible charge of AI-assisted work (ASCE, 2024; NSPE, 2024)[3], and licensing boards now require engineers to test AI tools against known answers and to keep auditable records of the tools, versions and validation steps behind each deliverable (PEGNL, 2026; NCBELS, 2025)[4].

Together, these forces create a paradox: AI promises more output from fewer people, but every extra option it produces adds to the checking load of exactly those senior people who are already scarce. Organizations that do not redesign verification will either drown their best engineers in review or stop reviewing properly and let persuasive errors through.

1.3 What This Paper Adds

This is my second paper in a series. My July and September 2026 papers in this journal reframed the project management office (PMO) as an AI-powered investment-protection system and found that D&E accounted for only 5.1% of 118 recent publications on AI in managing industrial projects (Lagutin, 2026a; 2026b). A companion paper, currently under review, asks *where* AI creates the most value; its answer is the front end, between FEL1/FEL2 and FEED[5]. This paper asks the next question: **how can the value created by AI be verified, approved and secured?**

It is written for three groups of readers (besides project managers and sponsors), those who play a near-critical role in creating project value: the heads and chief engineers of engineering organizations (EP, EPC and EPCM contractors, design institutes and OEM engineering divisions); BIM and digital leaders; and cost and value engineers. Owner's engineers, who must judge contractors' AI-assisted work, are the next important audience. The paper offers them a diagnosis of why verification now caps the value of AI (Sections 3 and 4), a method with named

[3]ASCE: American Society of Civil Engineers; NSPE: National Society of Professional Engineers (both US).

[4]PEGNL: Professional Engineers and Geoscientists Newfoundland and Labrador (Canada); NCBELS: North Carolina Board of Examiners for Engineers and Surveyors (US).

[5]Front-end loading (FEL) is the staged definition of a capital project before the investment decision: FEL1 covers scoping, FEL2 the selection of the preferred option, and FEL3, or front-end engineering design (FEED), the definition of the chosen option; detailed engineering and construction follow.

© 2026 Konstantin Lagutin

owners and tested tools (Section 5), three supporting disciplines (Section 6), new roles for familiar people (Section 7), an implementation kit (Section 8), and, in closing, what each group can start doing right away (Section 11).

2. Where the Evidence Comes From

The argument rests on three kinds of evidence. The first is a literature corpus: for this article, 220 sources published between January 2023 and September 2026 were screened and 134 kept, from journals, AACE International papers, engineering research, industry practice and professional bodies; their 324 problem mentions were merged into 25 open issues and ranked by frequency and estimated impact (Figure 1). The second is first-hand evidence from three programs in which I took part: a major East European integrated steelmaker (2013-2020; BIM and 4D/5D modeling), a global mineral-fertilizer producer (2023-2024; AI-supported value engineering (VE) on a greenfield phosphate plant in Central Asia) and a US$1bn+ Gulf green-steel program based on direct reduction of iron (DRI) (2024-2026; technology selection and an AI-enabled digital PMO). Company names are withheld. The third is a new scan of the industries that met the verification barrier earlier, of the safety-critical industries whose checking practices the method borrows, and of the newest cross-industry evidence on the value of agentic AI.

Figure 1 maps the 25 issues by demand (how often the sources raise them) and impact (their expert-judged effect on project duration and cost). Two dividing lines create four quadrants.

- Q1, "Act now", holds issues that are both widely discussed and costly.
- Q2, "Hidden value", holds costly issues that the literature still underrates, such as weak physics grounding, knowledge reuse and overdesign; here lie the least-contested gains.
- Q3, "Enablers", holds popular topics with an indirect effect.
- Q4, "Watch list", holds background risks to monitor.

The map explains the "1 + 3 + Enablers" formula of this paper: its core theme answers the verification bottleneck (Q1) and persuasive AI errors; three supporting topics answer the Q2 issues that verification depends on; and the Q3 enablers shape the implementation. Table 2 lists all 25 issues and, in its last column, where those addressed here are treated.

© 2026 Konstantin Lagutin

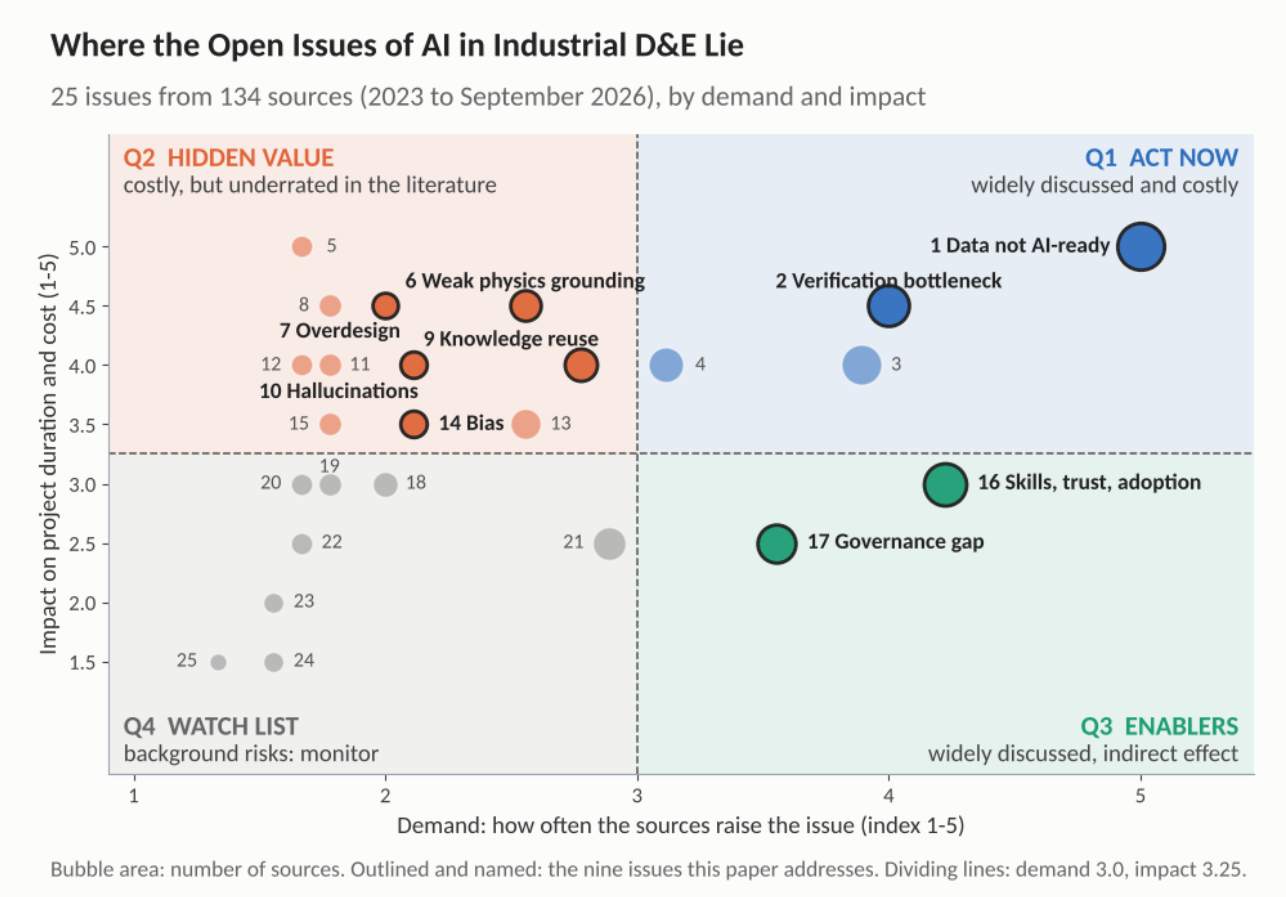


Figure 1. Twenty-Five Open Issues of AI in Industrial D&E: Demand vs. Impact

Table 2. Twenty-Five Open Issues of AI in Industrial D&E, by Quadrant and Impact

Matrix Quadrant	#	Issue	Sources	Impact (1-5)	Role: sections
Q1 Act now	**1**	**Data not AI-ready**	**36**	**5.0**	**Topic 2: 6.2**
	2	**Human verification bottleneck**	**27**	**4.5**	**Core: 4, 5, 7**
	3	Forecast accuracy	26	4.0	
	4	Toolchain integration	19	4.0	
Q2 Hidden value	5	AI applied too late	6	5.0	
	6	**Weak engineering (physics) grounding**	**14**	**4.5**	**Topic 1: 6.1**
	7	**Overdesign and unchallenged specifications**	**9**	**4.5**	**Topic 3: 6.3**
	8	Weak design-to-field feedback	7	4.5	
	9	**Knowledge reuse failure**	**16**	**4.0**	**Topic 2: 6.2**
	10	**Hallucinations and false confidence**	**10**	**4.0**	**Core: 4, 5**
	11	Engineering capacity crunch	7	4.0	
	12	Misaligned contracts and incentives	6	4.0	
	13	Scarce unique-project data	14	3.5	
	14	**Bias in data and models**	**10**	**3.5**	**Topic 3: 6.3**
	15	Missing life-cycle value models	7	3.5	

© 2026 Konstantin Lagutin

Q3 Enablers	**16**	**Skills, trust and adoption**	**29**	**3.0**	**Enabler: 7, 8**
	17	**AI governance and control gap**	**23**	**2.5**	**Enabler: 7, 8**
Q4 Watch list	18	Accountability and liability	9	3.0	
	19	Black-box outputs	7	3.0	
	20	Junior pipeline erosion	6	3.0	
	21	Unproven value	17	2.5	
	22	IP and data rights	6	2.5	
	23	Cybersecurity	5	2.0	
	24	Regulatory uncertainty	5	1.5	
	25	Implementation cost and vendor lock-in	3	1.5	

Source: author's issue map (see companion paper); quadrants as in Figure 1. Bold: addressed in this paper (role and sections in the last column).

Vendor and consultancy figures are given as their claims, and the illustrative calculations in Section 4 show ratios and mechanisms, not benchmarks.

3. The Question Has Changed from "Can AI Draw?" to "Can We Trust What It Drew?"

In the nearly four years since the public release of ChatGPT, the question that engineering leaders ask about AI has changed (Table 3).

Table 3. In 3.75 Years, the Frontier Moved from Generating to Checking

Topic	2023: starting point	2024-2025: what moved	2026: status quo and frontier	From the author's practice
Verification	AI outputs reviewed ad hoc; no roles or records	Engineer stays in responsible charge; cursory review of AI output is unethical (ASCE, 2024; NSPE, 2024)	Best schedule-quality agent matches experts on 74.9% of items (Jackson, 2026); boards require testing against known answers (PEGNL, 2026)	In 100+ projects, drawing and VE reviews were fully manual; AI was proposed as a "first-pass checker"
Physics	Chat interfaces with no engineering model behind them	Text2BIM agents correct models against 30 rules (Du et al., 2026); tool protocols let models drive simulators	Best model passes under 35% of engineering design tasks (Guo et al., 2025); agentic flowsheet design fails on multi-component systems (Schäfer et al., 2026)	At the steelmaker, software found the clashes; engineers closed each one
Memory	Lessons learned in folders; know-how in senior heads	Knowledge copilots enter production at large EPCs and OEMs	Source-citing retrieval recommended first (Sadeh et al., 2026); memory is the most value-relevant agent control (Apotheker et al., 2026)	The fertilizer producer analyzed decades of errors and claims before GenAI-supported VE
Overdesign	Safety factors and specifications	ML cost models make design-to-cost checks practical (Leitão et al., 2024)	GenAI screens a design for safety-factor inflation in about two minutes (Shellshear, 2026)	20+ AI-supported VE proposals cut a CAPEX estimate by 16.5%

© 2026 Konstantin Lagutin

	inherited, rarely challenged			

Machines learned to draw before they learned to check. In the NeurIPS 2025 engineering-design benchmark, no model passed more than 35% of 101 tasks in nine engineering domains, and 55-67% of failures came from missing domain knowledge and violated constraints; the authors had to build simulation-based evaluation, because comparing answers with a key could not tell good designs from bad ones (Guo et al., 2025). **Rule-based checking is old; making it affordable is new.** Automated compliance checking has been researched for four decades with "limited industry adoption", mainly because translating codes into machine-readable rules was too costly (TUM GNI, 2026); language models now help draft and explain such rules. **Professional rules have adapted as well.** The message changed from "the engineer stays responsible" to "show how you checked": guidelines now ask engineers to verify that an AI tool "performs reliably and correctly" (PEGNL, 2026) and that its results can be "fully understood, replicated, and, if necessary, reproduced by the licensee" (NCBELS, 2025). Figure 2 sums up the shift.

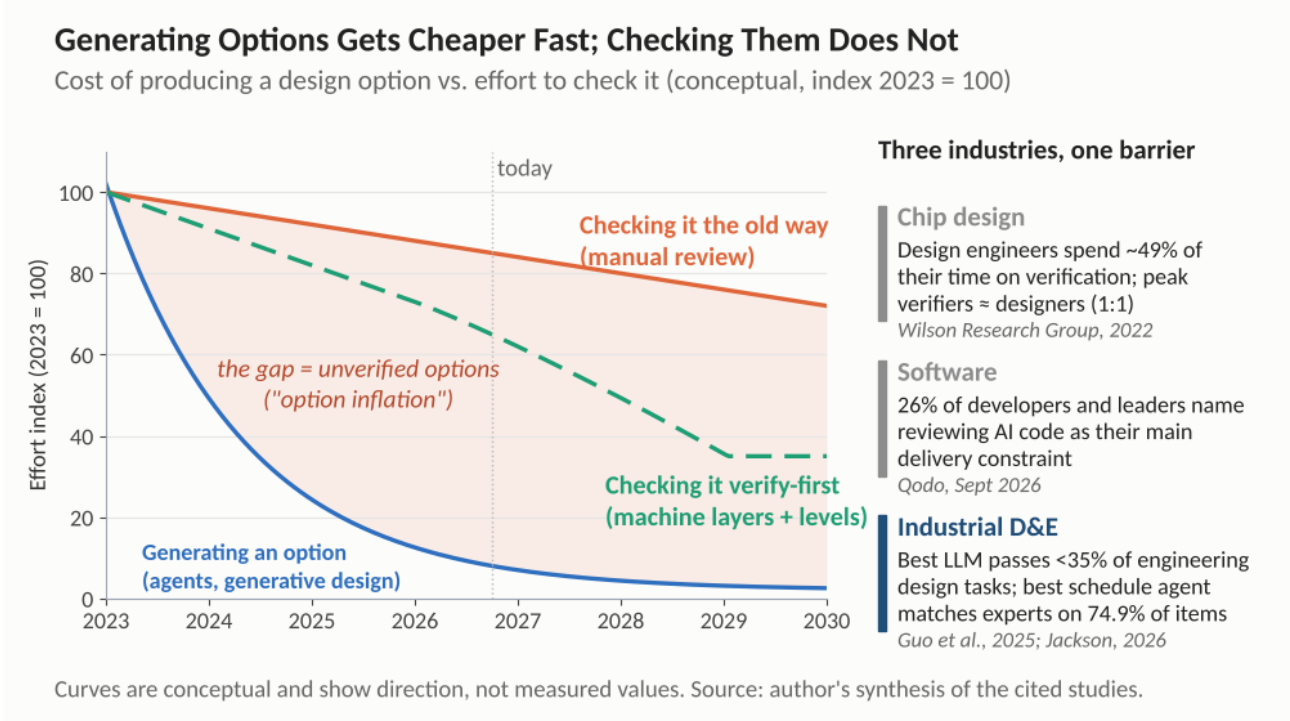


Figure 2. Generating Options Gets Cheaper Fast; Checking Them Does Not

Conceptual curves; evidence panels from Foster (2022), Qodo (2026), Guo et al. (2025) and Jackson (2026).

© 2026 Konstantin Lagutin

4. Verification, Not Generation, Now Caps the Value of AI

Why has checking, rather than generating, become the limit on value? Three well-established ideas from management science and safety engineering explain it. Box 1 summarizes them in plain terms.

Box 1. Three Ideas Behind the Method, in Plain Terms

- **Bounded rationality and satisficing** (Simon, 1955). People and organizations have limited time, information and attention, so they stop at the first option that meets their aspiration level: they *satisfice*. Decision quality is bounded by how much search they can afford. AI makes search cheap; the bound moves to checking.
- **Automation bias** (Parasuraman and Manzey, 2010). People over-rely on automated advice: they miss errors the system does not flag and follow wrong advice it gives. It grows when the automation is usually right and checking takes effort, and it affects experts too. The remedy lies in process design, not in appeals to be careful.
- **The Swiss cheese model of layered defenses** (Reason, 2000). Every barrier against error has holes. Failures happen when holes line up, so several different, imperfect barriers are safer than one that claims to be perfect. The verification stack in Section 5.2 applies this idea to AI output.

4.1 Search Is Now Cheap; Trust Is Not

Value engineering has always been bounded by the cost of looking. Lawrence Miles invented it in the 1940s at General Electric to ask what a part must do and how else that function could be delivered (Miles, 1961). Yet a VE workshop of 40 hours at 30% design, with a handful of senior people in the room, compares two or three alternatives, because that is all it can afford to examine: a textbook case of *satisficing*.

AI collapses the cost of search, but not the cost of belief. Each option that may change the way loads are carried through a structure, a mass balance or a hazardous area still has to be understood, tested and owned. When search was expensive, checking capacity was rarely the limit. When search is nearly free, it sets the new bound of rationality. **AI moves the bottleneck of engineering judgment from finding options to trusting them.**

4.2 Unmanaged Generation Destroys Value in Three Ways

Option inflation. Engineers receive more alternatives than they can examine. They either review them all, consuming the hours AI was supposed to free, or review a sample and let the rest

© 2026 Konstantin Lagutin

through on trust. Software teams report the same: 36% of developers say reviewing AI code takes as long as ever, but with greater cognitive effort (Qodo, 2026).

Persuasive errors. AI output is fluent, formatted and confident; wrong numbers arrive in the same font as right ones. Cost engineers call this "false confidence" (Patel and Kumbharathi, 2026).

Automation bias. A 2026 experiment with 2,784 participants verifying AI-extracted data found that prior attitudes toward AI were the strongest predictor of accuracy, and that when correcting an AI suggestion required more effort, participants made fewer corrections and accepted more wrong suggestions; bonuses did not help (Beck et al., 2026). The lesson: **verification quality is a design property of the well-established process, not a virtue of the reviewer.** If correcting an agent takes ten clicks and accepting it takes one, people will accept the AI's suggestion, right or wrong.

4.3 The Funnel: Let Machines Throw Away What Engineers Should Never Read

Figure 3 follows one VE round on one FEL2-FEED package, using illustrative numbers from the author's practice. Agents generate 60 options. Automated rule, code, clash and quantity checks remove more than half within minutes; simulators remove another dozen within hours; engineering memory and a red-team challenge remove four more. Only ten options reach human reviewers, who spend about 30 hours on them, scaled to risk, and accept eight.

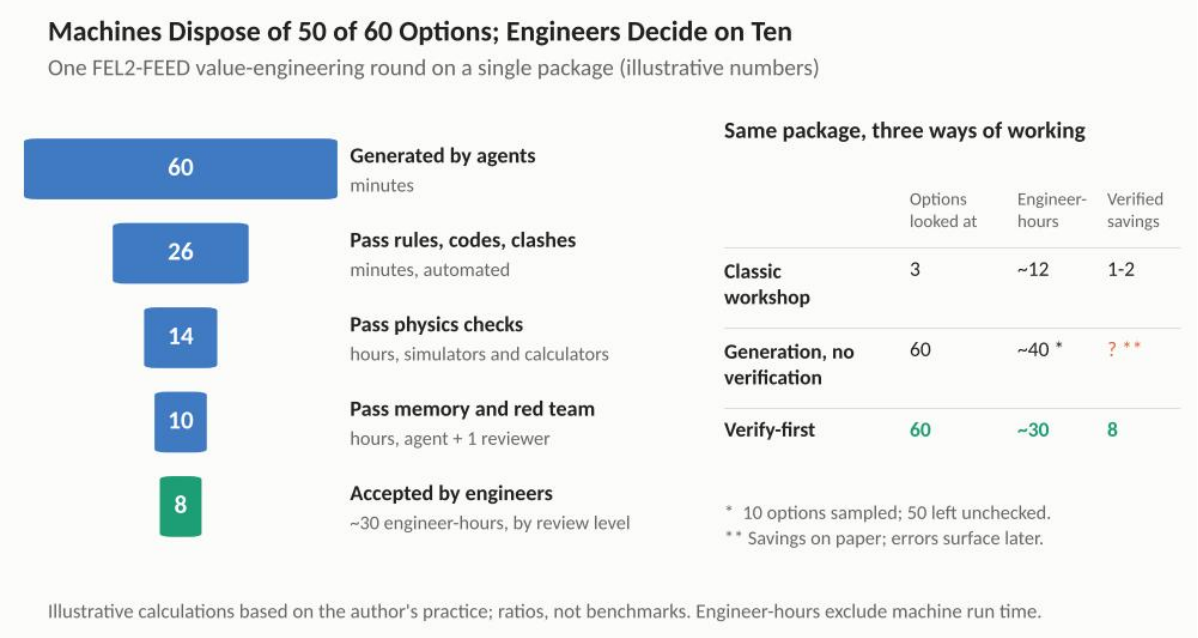


Figure 3. Machines Dispose of 50 from 60 Options; Engineers Decide on Ten

Review hours assume about 1 hour per level-2 item, 4 hours per level-3 item and 8 hours per level-4 item (Section 5.3).

© 2026 Konstantin Lagutin

The economics are clear. A classic workshop examines three options in about 12 engineer-hours and verifies one or two preferable feasible options. Generation without verification looks at 60 options, spends about 40 hours sampling ten and leaves 50 claims unchecked. The verify-first AI engine (Section 5) examines the same 60 in about 30 engineer-hours and produces eight verified savings. The right unit of account is therefore not cost per token or per drawing, but **cost per verified option**: machine compute plus the engineer-hours needed to check it. BCG recommends the same metric for agentic AI: cost per successful outcome, "including the human effort needed to review and operate the workflow", and deterministic software wherever a model is unnecessary (Apotheker et al., 2026). A more capable model that gets an option right the first time can be cheaper than a cheap one whose output engineers have to redo.

Two testable propositions follow. P1: the VE value an organization captures behaves like a product of *option breadth × verification strength × timing*; if any factor is zero, the result is zero. **P2:** increasing option breadth without increasing verification strength *raises* the number of errors found late, in detailed design, construction or commissioning. If verification strength is the factor that binds, the practical question is how to build it into everyday engineering work, with clear owners and without drowning the senior engineers. Section 5 proposes an answer.

5. The Verify-First Value Engine: Agents Propose, Checkers Dispose, Engineers Decide (AP/CD/ED)

The verify-first value engine is the AI-supported working process proposed here: GenAI agents generate options and routine deliverables, automated checkers dispose of unsound ones, and accountable engineers decide on the rest, with every claimed saving tracked until it is verified. It is recommended first of all to engineering directors and chief engineers who must scale AI without losing control of quality, to value and cost engineers who must prove savings, and to BIM and digital leaders who build the platform. This section explains how to make these six words work inside the industrial BIM (iBIM)[6] of an asset, who does what, how value is measured, and which parts are already proven.

[6]Industrial BIM (iBIM) is the object-based 3D/4D/5D information model of an industrial asset (a plant, a unit such as a blast furnace or DRI module, or an energy installation) with its engineering data and common data environment under ISO 19650.

© 2026 Konstantin Lagutin

5.1 Redraw the Job Plan Around Verification Gates

SAVE International's Job Plan[7] structures a value study in eight phases (SAVE International, n.d.). The verify-first engine keeps the phases but changes who does what in each, and it inserts three machine verification gates (V1 rules, V2 physics, V3 memory and red team) where options can still be discarded cheaply (Table 4).

Table 4. What Changes in Each Phase of the Job Plan

Phase (1-8)	Classic practice	With AI agents and verification	What the engineer must still do
1. Preparation	Choose the study scope; set up the workshop	Agent assembles the package from iBIM, schedule and cost data	Set the review level and the checking budget
2. Information	Team reads drawings and reports	Agent retrieves model data, precedents and error history, with citations	Confirm the design basis and constraints
3. Function analysis	FAST diagram on a whiteboard	Agent drafts the function diagram and cost-to-function split	Confirm the functions; no AI sign-off here
4. Creativity	Brainstorm 30-50 ideas, most never developed	Agents generate 50-100 options	Add ideas the agents missed
5. Evaluation	Scoring by vote	Gates V1 and V2 remove unsound options	Review survivors by risk level
6. Development	3-5 proposals written up	Agent details and prices survivors; gate V3 tries to break each saving	Accept, modify or reject, and record why
7. Presentation	Slides to the sponsor	Options with CAPEX, OPEX and CO_2 deltas linked to iBIM objects	Sign; submit to the value register
8. Implementation	Often lost after the workshop	Agent tracks adoption and the as-built result	Feed lessons back to engineering memory

VE stops being an event. A workshop happens once or twice per project; the engine can run on every package at every gate, because its marginal cost is mainly machine time. And senior judgment is spent only where machines cannot decide.

5.2 Stack Six Verification Layers, Fastest and Cheapest First, and Name Their Owners

Figure 4 applies the Swiss cheese model (Box 1): six layers, ordered from the fastest, cheapest and most explainable checks to the slowest, most expensive and most human ones. "Cheapest"

[7]SAVE International, founded in 1958 as the Society of American Value Engineers, is the US-based professional society for value engineering; its Job Plan is the standard sequence of a value study.

© 2026 Konstantin Lagutin

means the cost of running a check, not the value of what it catches: inexpensive machine layers remove most unsound options, so senior judgment is spent only where it matters.

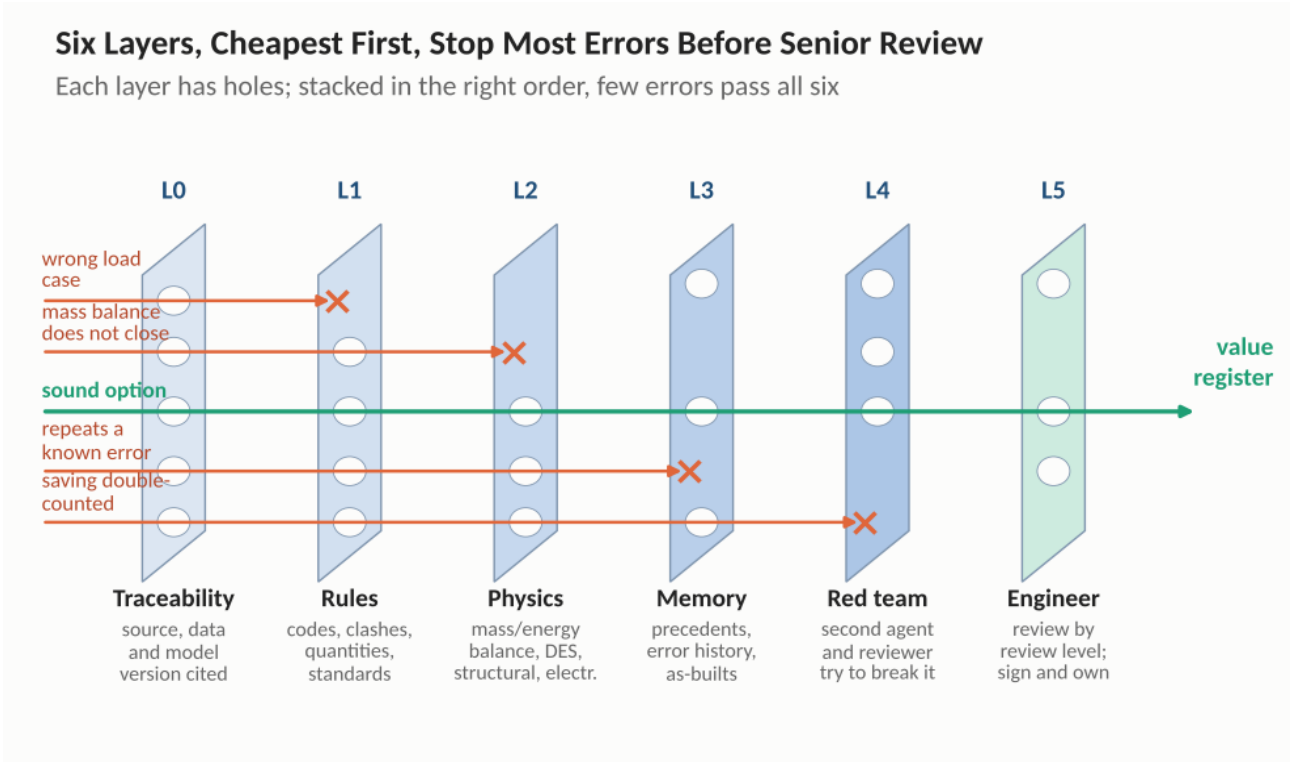


Figure 4. Six Layers, Cheapest First, Stop Most Errors Before Senior Review

Each layer catches a different class of error; engineers see only the survivors.

- **L0 Traceability.** Every number and claim carries its source (source traceability): model object, standard clause, precedent, calculator, data and model version. Untraceable output is rejected before anyone reads it.
- **L1 Rules.** Deterministic checks of codes, clashes, quantities and company standards: cheap, fast and already trusted. Findings travel back as BIM Collaboration Format (BCF)[8] issues attached to the objects concerned (Du et al., 2026).
- **L2 Physics.** Mass and energy balances, discrete-event simulation (DES)[9] of logistics, structural and electrical calculations, run by trusted engineering software that the agent calls (see Section 6.1).
- **L3 Memory.** The option is compared with precedents, error histories, non-conformance reports and as-built outcomes (Section 6.2).

[8]BCF is an open buildingSMART standard for exchanging model-based issues linked to the objects concerned.

[9]Discrete-event simulation models a system as a sequence of events (a ladle arrives, a crane is busy, a conveyor stops) to test throughput, queues and failures before anything is built.

© 2026 Konstantin Lagutin

- **L4 Red team.** A second agent and a value engineer try to break every claimed saving: Is it double-counted? Does it shift cost to operations? Does it void a vendor guarantee?
- **L5 Engineer.** An accountable engineer reviews the option at the depth its review level requires, accepts, modifies or rejects it, signs, and records the reasons.

A stack without owners is a diagram, not a process. Table 5 names who runs each layer, who maintains it and who accepts its result. Machines run the checks, but people own the rules, and the people who own the rules are the most experienced engineers, because the rules encode their judgment.

Table 5. Who Does What in the Verification Stack

Layer	Runs the check	Sets and maintains it	Accepts the result
L0 Traceability	The agent platform, automatically	Digital and BIM team	Checker operator rejects untraceable output
L1 Rules	Rule, code and clash engines	Discipline leads write rules; BIM manager configures; chief engineer approves	Checker operator triages; discipline engineer closes
L2 Physics	Simulators called by the agent; checker operator runs cases	Discipline specialists own the models	Discipline engineer
L3 Memory	Retrieval agent	Knowledge owner in QA/QC codes NCRs, claims and lessons	Discipline engineer
L4 Red team	Second agent plus a value or cost engineer	Value-engineering lead	Owner of the value register (see Section 5.5)
L5 Engineer	Discipline lead or engineer of record; review level 4 adds an independent second checker	Chief engineer sets the review levels and autonomy levels	Engineer of record signs

5.3 Matching Review Depth to the Stakes and Earn Autonomy Task by Task

Not every AI output deserves the same scrutiny. A document search and a change to the design basis of a furnace differ in what happens if they are wrong and in how easily the mistake can be undone. Figure 5 combines consequence and reversibility into four review levels.

© 2026 Konstantin Lagutin

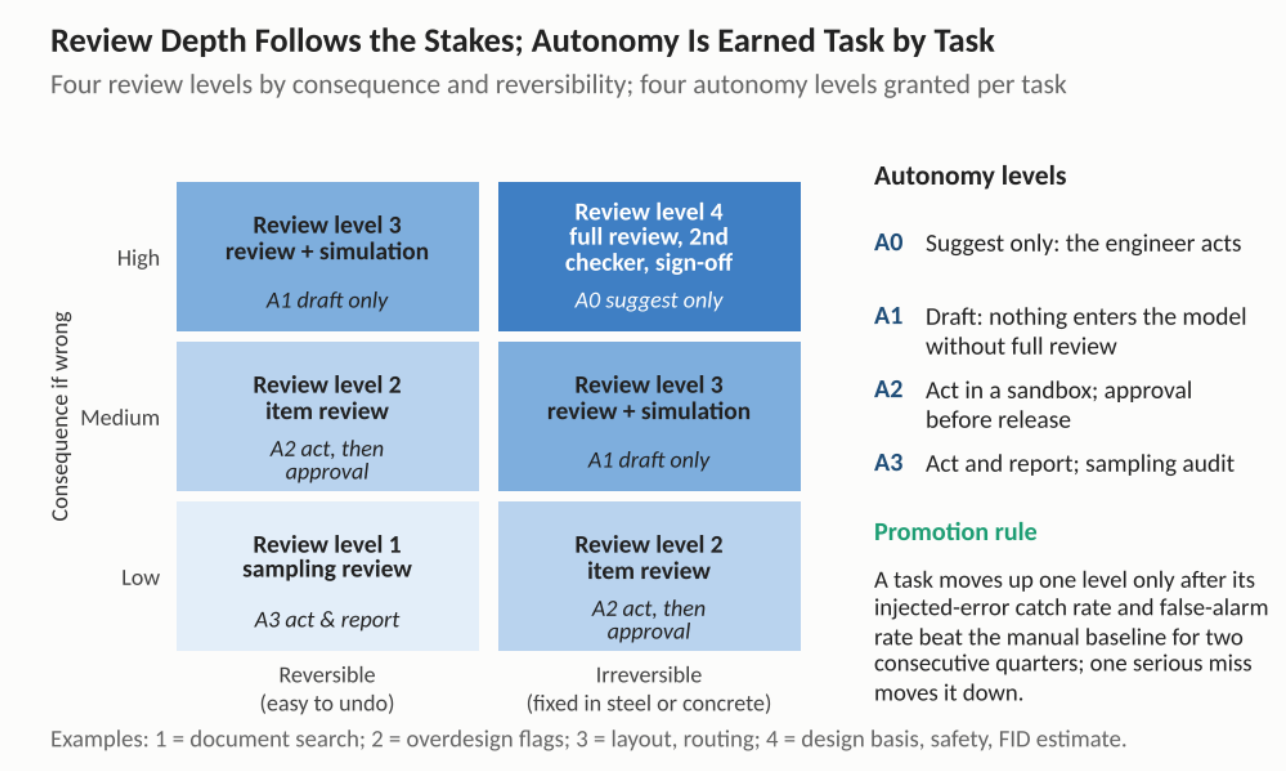


Figure 5. Review Depth Follows the Stakes; Autonomy Is Earned Task by Task

Autonomy levels A0-A3 are granted per task, not per tool, and can be withdrawn.

Autonomy is set at the task level, not the tool level. No agent is "approved" in general; each task earns its level with evidence. An agent may act and report (A3) on document search long before it may even draft (A1) a change to a safety system. A task moves up one level only after its catch rate on injected errors (Box 3) and its false-alarm rate beat the manual baseline for two consecutive quarters, and one serious miss moves it down again. Leading companies outside engineering do the same: a global business-process outsourcer built its operating model as a ladder from insights to copilot to autopilot and sets autonomy "at the process level, not at the level of individual tools" (Apotheker et al., 2026).

Two rules make the review levels work. *Budget checking hours like design hours:* if a package plan shows 2,000 design hours and no line for verifying AI-assisted work, checking will be squeezed out. *Make correction cheaper than acceptance:* require a short reason for *accepting* a level-3 or level-4 option, not only for rejecting it.

5.4 Put Six Controls Around Every Agent

Controls multiply value rather than limiting it. BCG found that companies with six agent controls in place enterprise-wide generate about three times as much value from agentic AI as companies with just one (1.8% vs 0.5% of revenue); only 5% of companies have all six (Apotheker et al.,

© 2026 Konstantin Lagutin

2026). The verification stack checks what agents *produce*; the six controls govern what agents *are allowed to do*. Table 6 translates them into engineering terms.

Table 6. Six Agent Controls, Translated for Design and Engineering

Control (relative importance to value)	What it means in D&E	What to require of platforms and vendors
Memory and context (32%, the largest)	Only verified lessons enter engineering memory; unverified agent output is quarantined so errors cannot propagate	Separate verified and unverified memory; source and version on every entry
Scope and oversight (21%)	Each agent has a written purpose, permitted actions and autonomy level per task	Permissions by task and autonomy level, not by user account
Evaluation & rollback (18%)	Release gates (fault-injection test, Box 3), drift monitoring, and rollback of any agent change to the model	Re-runnable proof records; model change history with one-step rollback
Tools and integration (13%)	A register of agents and their tools; open interfaces such as MCP[10] and BCF; no agent sprawl	Agent registry; open tool connectors and issue formats
Ownership (9%)	Every material agent has a named owner, usually a discipline lead, who can suspend or stop it	Owner field and a "stop" switch per agent
Security (7%)	Private deployment, runtime limits, audit trail, cost guardrails	Audit trail of inputs, tools, model version, reviewer and decision (PEGNL, 2026; NCBELS, 2025)

Controls and their relative importance after Apotheker et al. (2026); translation to D&E terms by the author.

The largest control is the one engineering firms think about least. Memory and context handling carries almost a third of the value effect, because memory compounds learning across agents but can also propagate errors. For a design office, this means a simple rule: **engineering memory accepts only what has passed verification.** An unchecked agent proposal that slips into the precedent library will be retrieved and repeated on the next ten projects.

5.5 Count Only Verified Savings

Controls keep the agents in bounds; the last element of the engine makes sure that their savings are real. **A saving that nobody independent has checked is a claim, not a result.** The verify-first engine keeps a **value register** with four stages, and a saving moves on only with named signatures and evidence (Table 7). The register belongs to the PMO's assurance function, staffed by cost and value engineers in project controls, not to the delivery team whose savings it records. As the Gulf green-steel program put it: delivery reports; assurance validates.

[10]Tool connections let an AI model call external software through a defined interface; the Model Context Protocol (MCP), introduced in 2024, is the widely adopted open standard.

© 2026 Konstantin Lagutin

Table 7. A Saving Moves Through Four Stages, Each with Its Own Signature

Stage	What is recorded	Who confirms	Evidence required
Claimed	Option, affected iBIM objects, estimated CAPEX, OPEX and CO_2 delta	Originating engineer or VE lead	Links to model objects and source data
Verified	Delta re-estimated on the baseline's basis, net of redesign and implementation cost, with no loss of function	Independent cost engineer in project controls, plus technical authority	Estimate revision, model-based quantities, L2 results, function check
Approved	The change incorporated in the approved design and budget	Change board or sponsor; owner's engineer co-signs	Approved change record, updated baseline
Realized	Saving confirmed at award or in final cost; energy and OPEX after 12 months	Project controls and finance; operations for OPEX	Contract prices, final cost, metered consumption

Tracking pays. Across sectors, companies that track AI value directly in the P&L report about three times the AI value vs. those that do not measure it (3.6% vs 1.2%), yet only 14% of CEOs have defined the P&L impact of every AI initiative (Apotheker et al., 2026). Four rules keep the register honest: 1) one iBIM object, one saving; 2) every saving net of its implementation cost and schedule effect; 3) life-cycle effects at agreed prices, including carbon; and 4) an annual audit sample. Public VE programs have long reported approved rather than proposed savings (FHWA, 2021); what is new is the link to model objects, which follows a saving from the first agent proposal to the as-built plant.

5.6 Most of the Tools Are Already Proven

The method is new as a whole; most of its parts are not (see Table 8).

Table 8. Tested Practice Behind Each Tool of the Verify-First Engine

Tool	Proven where	Evidence	Status in industrial D&E
Rule and clash checking (L1)	BIM coordination of capital projects	3,058 clashes found pre-construction (author's practice); rule-checker loop (Du et al., 2026)	Proven
Simulator-checked options (L2)	Process design and debottlenecking	AI copilot drives a process simulator (Kang et al., 2026)	Tools proven; AI orchestration piloted
Risk-graded verification	Nuclear facilities; medical devices	Checks scaled to safety significance (IAEA, 2016) and process risk (FDA, 2025)	Proven elsewhere

© 2026 Konstantin Lagutin

Independent check of high-consequence items	Functional safety	Assessor independence rises with safety integrity level (IEC, 2010)	Proven
Injected errors (fault injection)	Aviation security; software testing	Threat image projection in EU aviation security (European Commission, 2015); mutation testing (DeMillo et al., 1978)	New to D&E (Box 3)
Automated checks before human review	Software engineering	First-pass test rates up from under 50% to over 80% (Lee, 2026)	Being transferred
Agent controls enterprise-wide	Cross-industry	About 3x agentic value with all six controls (Apotheker et al., 2026)	New to D&E (Table 6)
Engineering memory feeding AI-supported VE	Process-plant design	20+ proposals accepted after checking; CAPEX estimate −16.5% (author's practice)	Tested
GenAI overdesign screening	Mining capital projects	Overdesign worth millions found in a three-month pilot (Shellshear, 2026)	Piloted
Staged value register	Public VE programs	Approved, not proposed, savings reported (FHWA, 2021)	Proven; iBIM link new
Graduated autonomy by task	Road vehicles; business processes	Levels of driving automation (SAE International, 2021); insights-copilot-autopilot ladder (Apotheker et al., 2026)	Proposed by analogy

Vendor and consultancy results are given as reported.

The balance is reassuring. Seven of the eleven tools are proven in engineering or in other industries, three have been tested or piloted on capital projects, and only one is a proposal by analogy. The value engine asks engineering organizations to assemble known parts in a new order. Box 2 shows them working together.

Box 2. The Engine at Work in a DRI Plant Study (Composite Example)

At FEL2 of a hydrogen-ready DRI plant, the owner wants to know whether the plot can be compacted, the process-gas heater downsized and the existing feed-stock storage bypassed at minimum cost (demolition is not an option). An agent reads the iBIM objects, heat and mass balance and design basis, and proposes 48 options.

- **L0-L1 (minutes):** 21 options lack a traceable source, break spacing rules for hydrogen safety zones, or violate clearance and tie-in rules around the feed-stock storage, which must stay in operation.

© 2026 Konstantin Lagutin

- **L2 (hours):** a process simulator closes the balances for each survivor; 11 fail, most because a smaller heater cannot reach reduction temperature after the switch from natural gas to hydrogen, or because the demolition of the storage yard remains a problem: discrete-event simulation shows that their bypass conveyors cannot carry the required feed rate while the yard stays in service.
- **L3-L4 (hours):** memory shows that a similar gallery shortening caused spillage on an earlier project, and a tie-in next to live storage forced two shutdowns; the red team finds two savings that move cost into operating spares. Five options are removed.
- **L5 (about two days):** engineers accept seven of the remaining 11, including a bypass that avoids demolition, three after modification, and log reasons for four rejections.

The seven enter the value register as *claimed*; the cost engineer re-estimates them to move them to *verified*. Only verified lessons enter memory.

6. Three Disciplines Make the Engine Work

Section 5 described the engine; three disciplines make it work in practice. Each answers one of the supporting issues in Table 2: physics in the loop keeps the numbers honest (Topic 1), engineering memory tells checkers where errors hide (Topic 2), and checks for excess turn verification from a cost into a source of savings (Topic 3). All three are tasks for chief engineers and discipline leads rather than for the IT department.

6.1 Physics in the Loop: Agents as Orchestrator, Not Oracle

Numbers come only from calculators, never from the language model's memory. General-purpose LLMs do not know thermodynamics, codes or constructability; they know how engineers write about them. The agent sets up, runs and reads trusted engineering software through tool connections; the software's job is to be right. When BASF researchers let agents design chemical flowsheets, the agents solved textbook cases but failed on multi-component systems and could not repair a mass balance that did not close (Schäfer et al., 2026). A copilot that drives a process simulator through tool connections keeps the engineer in charge and the physics intact (Kang et al., 2026).

In metals and mining, physics in the loop has familiar faces: heat and mass balances for a pellet plant's induration furnace or a DRI shaft furnace's gas loop; discrete-event models of hot metal, slab or concentrate flows; thermal models of a reheating furnace; load-flow studies for an electric arc furnace (EAF) shop. An engineering transformation program designed in 2026 for a European metals-technology OEM made the point bluntly: a proposal copilot is worthless unless

© 2026 Konstantin Lagutin

its numbers are checked against the mass balance and a process simulator. Organizations that fund chat pilots instead of simulator connections are buying fluency, not engineering.

6.2 Engineering Memory: The Checker's Knowledge Base and the Firm's Cortex

Experienced reviewers know where errors hide; memory makes that knowledge available to every checker. Engineering memory is the governed, searchable record of what the organization has historically learned: error histories, non-conformance reports (NCRs), change orders, claims, as-built deviations, commissioning problems, standard solutions and the reasons why options were rejected.

For example, the fertilizer producer in my practice analyzed for decades of its own design errors and claims before using GenAI to support VE on its greenfield phosphate complex. That history gave the AI tools precedents no public model could know, and told reviewers where to look first. More than 20 AI-supported VE proposals followed; after engineers checked them, all were accepted, many in modified form, and the CAPEX estimate of US$1,283 million fell by US$211.7 million, or 16.5%.

Three practices turn memory into a working checker. *Code errors by object type,* not by project: "reheating furnace, skid-pipe support, thermal expansion" is retrievable; something like "Project 17, lessons learned.docx" is not. *Retrieve with citations,* the recommended first step of GenAI adoption (Sadeh et al., 2026). *Turn overrides into lessons:* the reason for every rejected or modified AI option goes into a structured field, and, under the memory control in Table 6, only verified lessons are admitted.

Memory is also where the strategic value lies. Every competitor can license the same models and platforms; nobody can license another firm's trial and error history. The designers of the OEM program put it in five words: **models are replaceable, memory is not.** BCG warns of "cognitive lock-in" and advises keeping the "enterprise cortex" (proprietary knowledge, business rules, decision logic and operating context) in a governed layer under its exclusive control, "technically separable and portable, not merely contractually owned" (Apotheker et al., 2026). For an engineering firm, that cortex is its checking rules, simulator models and coded error history.

6.3 Verification in Reverse: Check for Excess, Not Only for Errors

Classic checking asks whether something is strong, large or safe *enough*. Value engineering also asks what is stronger, larger or more conservative than it needs to be. AI makes this verification in reverse cheap enough to run on every package. Overdesign hides in plain sight: a Cambridge study of real steel-framed buildings found that the average beam used less than half of its load capacity (Moynihan and Allwood, 2014), and industrial plants stack margins from process

© 2026 Konstantin Lagutin

engineers, structural engineers and suppliers, each reasonable alone. A GenAI tool deployed at a mining company screens a design for safety-factor inflation and material escalation in about two minutes and, in a three-month pilot, found overdesign worth millions of dollars, including issues missed by earlier expert reviews (Shellshear, 2026). Table 9 lists typical signatures.

Table 9. Overdesign Leaves Recognizable Signatures

Signature	What the check does	Example in metals and mining
Stacked safety factors	Traces each margin to its owner and code clause; flags undocumented excess	Crane girders sized for a crane two classes above the duty
Inherited specifications	Compares each specification with the design basis and precedents	Alloy grade copied from a more corrosive service
Redundant equipment	Compares redundancy with reliability data	N+2 pumps where history supports N+1
Oversized utilities	Checks connected load against simultaneous demand	Transformers sized on nameplate sum, not diversified load
Material escalation	Benchmarks tonnage per unit of capacity against a reference class	Steel tonnage per tonne of pellet capacity above peers

The same machinery exposes bias in historical data: overconservative standards on one side, optimistic schedules and estimates on the other. A challenge log records every assumption tested against an independent reference class. But removing margins without understanding them is gambling, not value engineering: some protect against what the model does not see, such as ore variability or future hydrogen operation. A margin is removed only when an accountable engineer has understood why it was there.

7. Reinvent the Engineering Workforce Around Humans and Agents

Physics, memory and checks for excess all depend on people who know how to use them. **Value comes mainly from people and processes, not algorithms.** BCG's "10-20-70" rule holds that leading companies spend about 10% of their AI effort on algorithms, 20% on technology and data, and 70% on people, organization and processes; AI leaders, the top 7.5% of AI performers, are also 3.2 times as likely as laggards to prepare for agentic AI through strategic workforce planning (Apotheker et al., 2026). A verify-first program should be budgeted the same way, and it starts with new roles for familiar people.

7.1 Give Familiar People New Roles

- **The chief engineer becomes the chief verifier.** Chief engineers are often seen as the brake on AI adoption, for good reason: their stamp is on the result. The engine gives them

© 2026 Konstantin Lagutin

the controls instead: the checking rules, the defined review levels and the promotion of tasks between autonomy levels. Thus, the brake becomes the steering wheel.

- **Junior engineers get a new entry role: checker operator.** AI is taking over the drafting, take-offs and routine calculations through which young engineers traditionally learned their profession. Juniors instead run the automated checks and simulations, sort the findings, discard false alarms and prepare real issues for senior review. In a year, a checker operator sees hundreds of options and learns why most of them fail, much as pilots build judgment in simulators, and keeps a career path open into the senior roles that verification needs.
- **Value engineers become portfolio managers of options,** managing the funnel, red-teaming claims and preparing entries for the value register.
- **BIM managers become stewards of the digital thread and owners of the agent register:** agents can act only on objects with stable identities, and someone must know which agents exist and what they may touch.

Adoption is built, not bought. A global technology company found that one 1,000-person engineering team gained only 10% from AI because the tools were used rarely and superficially; six-week coaching sprints raised the share of power users from 5% to 50%, and these engineers became about 80% more productive (Apotheker et al., 2026). Engineering offices should coach checking with the same discipline.

7.2 Design the Process Against Automation Bias

Training alone will not protect reviewers; the process must. **Injected errors in live work:** a few known errors are planted in AI-assisted packages throughout the year, as airport screeners see fictional threat images among real bags; reviewers stay alert, and the catch rate becomes a live indicator. **Blind review for level-4 items:** the second checker does not see the first checker's conclusion or the agent's confidence score. **Reasons for acceptance,** not only rejection. **Rotation** between packages, so that familiarity with one agent does not turn into trust in it.

7.3 A Special Benefit: Reinvest Released Hours in Value Engineering

This is a special benefit of the verify-first approach: small routine savings pay for large front-end savings. AI releases senior hours from routine work, but released hours quietly disappear into other tasks unless they are deliberately reassigned. The verify-first organization reserves a fixed share of them, for example half, for verification and VE at FEL and FEED. The modest but certain gains of routine automation (a few percent of TIC) thus fund the checking capacity that unlocks the much larger gains at the front end, where most of a project's cost is still open to change.

© 2026 Konstantin Lagutin

8. Five Steps to Put the Engine to Work

Sections 5 to 7 describe what the engine is and who runs it. This section turns them into an implementation kit in five steps: map where GenAI and the verifiers sit in the workflow (8.1), build the architecture (8.2), concentrate on a few workflows with stop rules (8.3), prove verification before scaling generation (8.4), and measure verified trust rather than hours saved (8.5).

8.1 Map Where GenAI Works and Where Verifiers Check

Figure 6 places the engine in a standard D&E workflow for one design package. Engineers set the design basis, review level and checking budget (step 1); agents retrieve data and precedents, draft routine deliverables and generate VE options (steps 2-3); machine verifiers apply layers L0 to L4 and return rejected options as BIM Collaboration Format (BCF) issues, which checker operators triage (steps 4-6); engineers review the survivors by review level, run the inter-discipline check, sign and submit savings to the value register (steps 7-8); and agents track adoption in the iBIM and feed field and as-built data back to memory (step 9). **GenAI works in three places, verifiers in three, and engineers keep every decision that commits money or safety.**

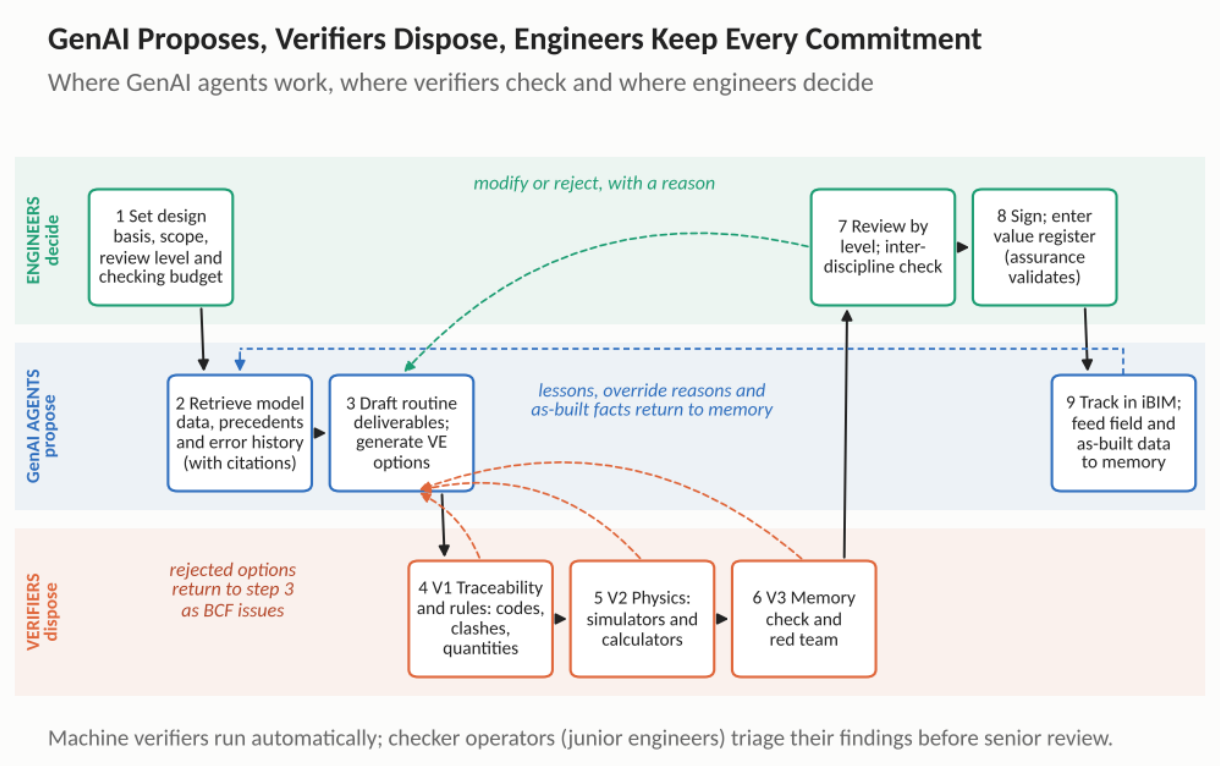


Figure 6. GenAI Proposes, Verifiers Dispose, Engineers Keep Every Commitment
Dashed arrows: feedback of rejected options, engineers' reasons and lessons to memory.

© 2026 Konstantin Lagutin

8.2 Build an Agent-Ready Architecture with Central Rules and Decentralized Teams

Figure 7 proposes an architecture for the AI-supported block. **Systems of record** stay where they are: the iBIM, engineering database, schedule, cost, documents, memory and field data. A **data thread** connects them: a common data environment under ISO 19650, stable object IDs and classification, a knowledge graph[11], tool connectors and issue exchange in the BIM Collaboration Format (BCF). **GenAI agents** under an orchestrator draft, generate and price options. **Verification services** run layers L0 to L4. The **engineer workbench** holds the L5 review console, the accept-modify-reject record, the value register and controlled write-back to the iBIM. Above them, **governance** holds the agent register, autonomy levels, fault-injection harness, scorecard and checking rules; below them, the **audit trail** records every action.

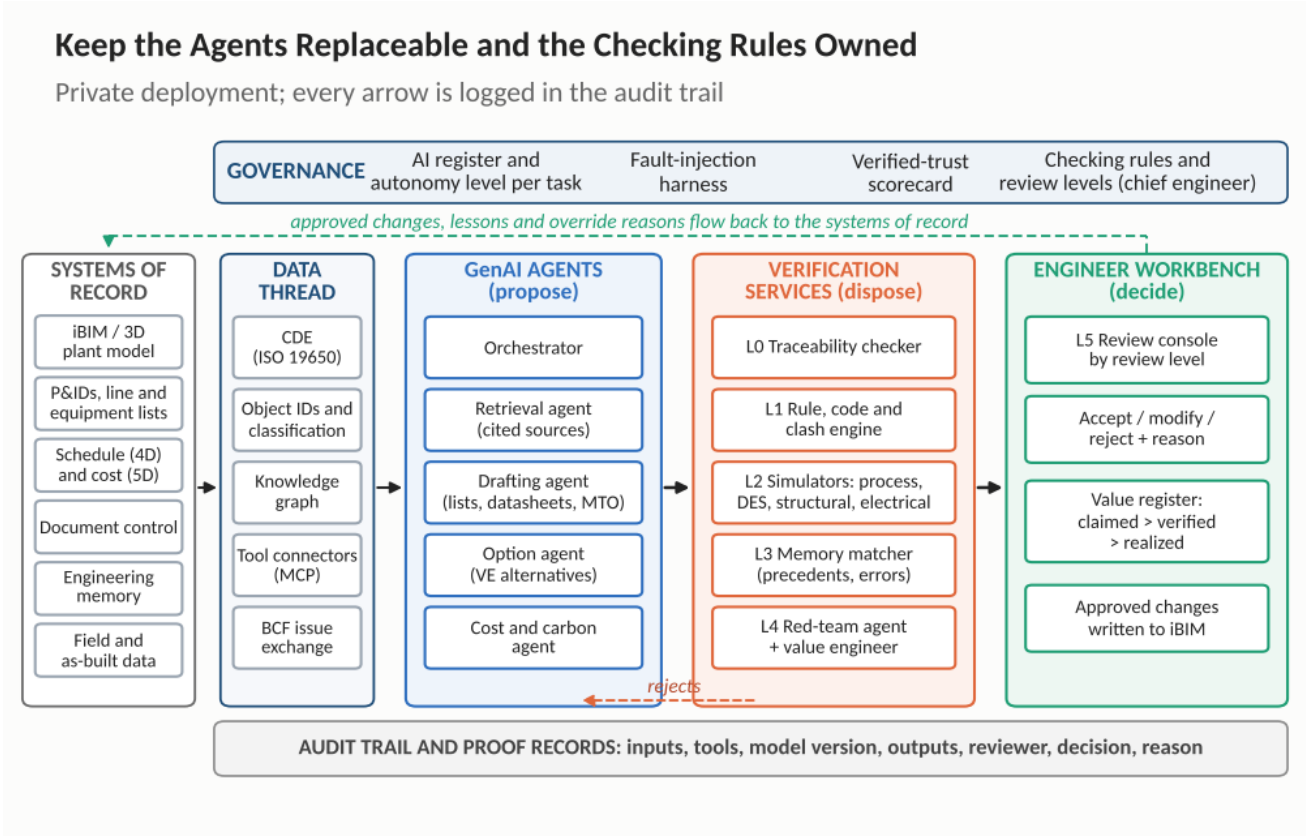


Figure 7. Keep the Agents Replaceable and the Checking Rules Owned

Tool types, not products; private, access-controlled deployment.

[11] A knowledge graph stores project data as linked objects (equipment, activities, cost items, inspections) with stable identities, so any system or agent can trace their relationships.

© 2026 Konstantin Lagutin

Three design choices deserve an executive decision.

Buy the agents, own the checkers: models commoditize quickly and should be replaceable across providers, whereas checking rules, simulator models and engineering memory (a structured, searchable repository of lessons learned, error histories and as-built data, in effect the firm's engineering knowledge warehouse) are the firm's cortex.

Set rules centrally, implement locally: companies that combine central risk and control guardrails with decentralized implementation bring three times as many workflows to scale as those that steer everything centrally (Apotheker et al., 2026); in a design firm, the chief engineer's office owns the rules and the six controls, and discipline teams build and run their own checkers.

Require the missing layer from vendors: platforms can prove that two pipes do not collide, but not that an agent complied with a design constraint, and even the 2026 draft revision of ISO 19650-1 does not mention AI agents (Day, 2026). The right column of Table 6 is the requirement list.

Owners can use the same list as a test of a contractor's maturity: instead of asking bidders which AI tools they use, ask **"Show us your verification model."** A contractor who can answer is selling verified value; one who cannot is selling beautifully formatted risk.

8.3 Concentrate on a Few Workflows and Run Experiments with Stop Rules

Depth beats breadth. AI leaders run five times as many AI workflows at scale as laggards (20 vs 4) from a portfolio only 1.5 times as large, and reach P&L impact in about 11 months rather than 17 (Apotheker et al., 2026). For an engineering firm, the lesson is to pick three routine operations from Table 1 and one VE lever, take them to scale, and avoid a "pilot museum". Good experiments run on live work, compare against a measured baseline and define in advance the result that would stop them (Table 10). The most important one is the fault-injection test (Box 3).

Table 10. Barriers, Remedies and Experiments with Stop Rules

Barrier	Remedy	Experiment, KPI and stop rule
Review capacity and automation bias (#2)	Review levels; checking budgets; independent check for level 4	Fault-injection test (Box 3); stop if the loop catches fewer errors than classic review
Persuasive errors (#10)	Source traceability mandatory; numbers only from calculators	Audit 50 AI-produced numbers; target 100% traceable
Weak physics (#6)	Simulators and codes as tools the agent must call	One lever with DES in the loop; KPI: options surviving review
Memory not usable or polluted (#9, #1)	Coding by object type; cited retrieval; verified-only admission	Code five years of NCRs and claims for one asset type; KPI: repeat-error rate

© 2026 Konstantin Lagutin

Inherited conservatism (#7, #14)	Challenge log for every margin above code	Overdesign scan of three packages; KPI: verified tonnage saved within code

Box 3 sets out this test step by step. It borrows fault injection, a long-established practice of software and safety engineering in which known errors are planted on purpose to measure whether a checking process catches them, and turns it into a go/no-go gate: no generative tool is scaled until its checking loop has shown, on a real package, that it catches errors at least as well as classic review.

Box 3. The Fault-Injection Test: A Go/No-Go Gate Before Scaling Any Generative Tool

1. **Choose a real package**, assisted by AI, for example a conveyor transfer station or furnace auxiliary system.
2. **Plant 20 known errors across five classes:** a wrong load case or design condition; a missing interface; a duplicated or missing quantity; a violated code or standard; an implausible or double-counted saving.
3. **Run two reviews in parallel:** the verify-first loop and the classic review, by people who do not know which errors were planted.
4. **Measure four numbers:** catch rate per class; false alarms; engineer-hours; time to close each finding.
5. **Decide:** scale only if the loop catches at least as many errors as the classic review, in fewer hours, in every class that matters for the package's review level. Otherwise, fix the checkers, not the generator.
6. **Repeat quarterly** with fresh errors, and after every model or tool change.

8.4 Prove Verification Before Scaling Generation

Figure 8 spreads the implementation program of an engineering organization over 24 months, about two annual budget cycles, long enough to build the checkers and prove them on live packages and short enough to keep the attention of management, with gates at months 6 and 15. Horizon 1 builds the cheap layers, the first agent controls and the baseline on one package; horizon 2 connects simulators, memory and the audit trail; horizon 3 scales the engine across FEL2 and FEED packages and grants autonomy where the evidence supports it.

© 2026 Konstantin Lagutin

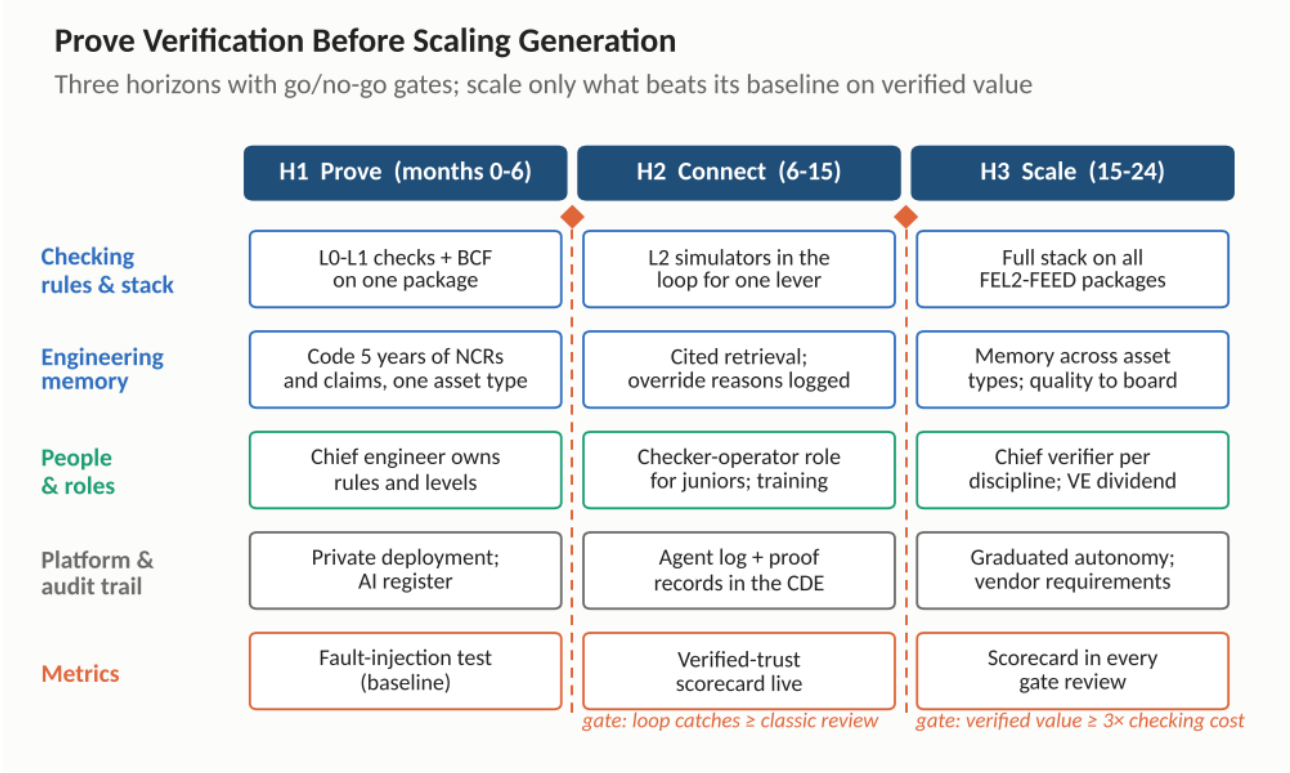


Figure 8. Prove Verification Before Scaling Generation

Each box is a candidate experiment; tools that fail their gates are retired.

8.5 Measure Verified Trust, Not Hours Saved

As follows from the above, **hours saved says nothing about whether anything trustworthy was produced.** Table 11 proposes a scorecard of verified trust. Its first readings will be uncomfortable, and usefully so: they show where the organization believes more than it has checked.

Table 11. The Verified-Trust Scorecard

Indicator	What it shows	Direction
Injected-error catch rate, by error class	Strength of verification	Up
False-alarm rate of the checkers	Whether checkers waste engineers' time	Down
AI-produced numbers traceable to a source or calculator	Traceability discipline	100%
Cost per verified option (compute plus checking hours)	Economics of the funnel	Down
Share of claimed savings that reach "verified" and "realized"	Honesty of the value register	Tracked
Repeat-error rate on new packages	Whether memory works	Down

© 2026 Konstantin Lagutin

Agents with a named owner, scope and all six controls	Control coverage	100%
Tasks at each autonomy level	Earned trust	Up, by evidence

9. Facing the Red Team

Before turning to the outlook, the method should face the same challenge it applies to every AI proposal. Below are the four objections most often heard from executives and chief engineers, each with a short answer.

"Verification will be automated too, so the bottleneck will disappear." Partly true: the stack automates most checking. But automated checkers also need checking, and accountability stays human. Chip design automated much of its verification and still spends about half of its engineering effort on it.

"We cannot afford to check 60 options." You do not check 60. Machines dispose of most of them; engineers review a handful, by review level.

"Controls will slow us down." The evidence says the opposite: enterprise-wide agent controls roughly triple agentic value (Apotheker et al., 2026). Controls are what allow autonomy to grow.

"Our chief engineer will block it." Then give the chief engineer the rules. In every program I have seen, adoption followed when the loop saved the reviewers' time rather than adding to it.

10. Outlook to 2030: Four Bets

What does all this mean for the next three to five years? The four bets below are forecasts, not certainties. Each is stated so that it can be proven wrong, and each tells a different audience what to prepare for: engineering leaders, professional bodies, platform vendors, and lawyers and insurers.

1. **Verification hours will be budgeted like design hours.** By 2029, large EPCs and design institutes will plan explicit checking hours for AI-assisted deliverables, and the ratio of verification to design effort will move toward the one-to-one that chip design reached. *Wrong if* package plans still show no separate checking line.
2. **Professional guidance will follow the licensing boards.** Within three to five years, value-engineering and cost-engineering bodies will publish recommended practices for AI-assisted VE and checking. *Wrong if* guidance stays limited to general statements of responsibility.

© 2026 Konstantin Lagutin

3. **Platforms will ship the missing layer.** Major design platforms will offer agent registries, audit trails, proof records and task-level permissions, and the next revision of ISO 19650 will likely address AI agents. *Wrong if* these remain custom add-ons by 2029.
4. **Disputes will turn on proportionate verification.** The first significant disputes over AI-assisted designs will be decided on whether verification was proportionate to the risk, not on whether AI was used. *Wrong if* courts and arbitration tribunals in construction disputes treat AI use itself as the fault.

> *Firms that keep buying generation without verification are buying beautifully formatted risk. The winners will not generate the most options; they will prove the most options, fastest.*

11. Conclusion: Build Verification Procedures First, Then Harvest the Bonanza

Generative and agentic AI have opened a bonanza in the routine, labor-intensive and error-prone work of industrial D&E, and they have solved the problem that limited value engineering for 80 years: the cost of looking. They have also created a new one: the cost of believing. This paper has argued that the organizations that capture the gains will build verification first: layers that dispose of unsound options cheaply, owners for every layer, controls around every agent, engineers who decide where judgment is needed, memory that admits only what was verified, and a value register that counts only what has been checked. Most of the tools are already proven. What is needed now is the decision to assemble them, and that decision does not have to wait for a large program. Table 12 sums up what each group of readers can take away and start doing on Monday.

Table 12. What Each Reader Can Take Away and Start on Monday

Reader	Key takeaway	What to start on Monday
Heads of EPC, EPCM and design organizations	Verification, not generation, now caps AI value; fund it like design	Pick three routine operations (Table 1) and one VE lever; budget 10-20-70; add a checking line to every AI-assisted package plan
Chief engineers and discipline leads	Owning the checking rules means steering AI, not braking it	Own the checking rules, review levels and the six agent controls; require reasons for overrides and level-3/4 acceptances
Value and cost engineers	VE becomes continuous, and only verified savings count	Redraw the Job Plan with verification gates; set up the four-stage value register in project controls
BIM and digital leaders	Agents are replaceable; rules and memory are the lasting asset	Build L0 and L1 first, on stable object IDs; set up the agent register; write Table 6 into platform requirements

© 2026 Konstantin Lagutin

OEM engineering leaders	Your coded error history is an advantage no competitor can license	Code your error history by object type before building proposal copilots
Owner's engineers	A verification model tests contractor maturity better than a tool list	Ask bidders for their verification model, not their tool list

None of this requires waiting for better models. It requires engineers to do what the best of them have always done, to check before they trust, and to extend that discipline to their new digital colleagues. I hope this article helps engineering leaders, chief engineers, value and cost engineers, BIM teams and owners turn the AI bonanza into verified value, and I wish every team that starts on Monday a successful first package, a few usefully uncomfortable scorecard readings, and many verified savings.

AI Use Disclosure

AI-enabled tools assisted with literature search, source registry organization, statistical analysis, and ranking. The author selected and verified sources, developed the concept, plan, and argumentation, wrote the main text content, and bears full responsibility for the accuracy and reliability of the work.

References

American Society of Civil Engineers (ASCE) (2024). Policy Statement 573: Artificial intelligence and engineering responsibility, 18 July. https://www.asce.org/advocacy/policy-statements/ps573

Apotheker, J., Beauchene, V., de Bellefonds, N., Bolden, D., Duranton, S. et al. (2026). *The Formula for Agentic AI Value: The Applied AI Index 2026*, Boston Consulting Group, September. https://www.bcg.com

Beck, J., Eckman, S., Kern, C. and Kreuter, F. (2026). Bias in the loop: How humans evaluate AI-generated suggestions, *Harvard Data Science Review*, April. https://hdsr.mitpress.mit.edu/pub/nrcn4h7d

Day, M. (2026). Agentic BIM's missing infrastructure, *AEC Magazine*, 28 April. https://aecmag.com/ai/agentic-bims-missing-infrastructure/

DeMillo, R. A., Lipton, R. J. and Sayward, F. G. (1978). Hints on test data selection: Help for the practicing programmer, *Computer*, 11(4), 34-41.

Du, C., Esser, S., Nousias, S. and Borrmann, A. (2026). Text2BIM: Generating building models using a large language model-based multiagent framework, *Journal of Computing in Civil Engineering*, 40(2), 04025142. https://doi.org/10.1061/JCCEE5.CPENG-6386

European Commission (2015). Commission Implementing Regulation (EU) 2015/1998 laying down detailed measures for the implementation of the common basic standards on aviation security, *Official Journal of the European Union*, L 299, 14 November.

© 2026 Konstantin Lagutin

Federal Highway Administration (FHWA) (2021). Value engineering summary report, FY 2021, U.S. Department of Transportation, Washington, DC. https://www.fhwa.dot.gov/ve/

Fjeldstad, S., Katsap, N. and Parbhoo, P. (2024). Reimagining engineering to deliver more projects, more efficiently, McKinsey & Company, 20 February.

Food and Drug Administration (FDA) (2025). Computer software assurance for production and quality system software: Guidance for industry and Food and Drug Administration staff, September.

Foster, H. (2022). Part 8: The 2022 Wilson Research Group functional verification study, *Verification Horizons*, Siemens EDA, 12 December. https://blogs.sw.siemens.com/verificationhorizons/2022/12/12/part-8-the-2022-wilson-research-group-functional-verification-study/

Guo, X. et al. (2025). Toward engineering AGI: Benchmarking the engineering design capabilities of LLMs, *Advances in Neural Information Processing Systems (NeurIPS 2025), Datasets and Benchmarks Track*. https://arxiv.org/abs/2509.16204

International Atomic Energy Agency (IAEA) (2016). *Leadership and Management for Safety*, General Safety Requirements No. GSR Part 2, Vienna: IAEA.

International Electrotechnical Commission (IEC) (2010). IEC 61508-1: Functional safety of electrical/electronic/programmable electronic safety-related systems, Part 1: General requirements, 2nd ed., Geneva: IEC.

Jackson, M. (2026). Beyond metrics: Utilizing AI agents for construction schedule quality analysis, AACE International Conference & Expo 2026, DSAA-4875.

Kang, J.-L., Yao, Y. and Wong, D. S. H. (2026). LLM-powered copilot for Aspen Plus: Enabling human-AI collaborative process design, *Industrial & Engineering Chemistry Research*, 65(31), 16718-16738. https://doi.org/10.1021/acs.iecr.6c00963

Lagutin, K. (2026a). From PMO to AI-powered investment protection system, *PM World Journal*, Vol. XV, Issue VII, July. https://pmworldjournal.com/article/from-pmo-to-ai-powered-investment-protection-system

Lagutin, K. (2026b). AI across the industrial project value chain: From tools to decision assurance, *PM World Journal*, Vol. XV, Issue IX, September. https://pmworldlibrary.net/wp-content/uploads/2026/09/pmwj168-Sep2026-Lagutin-AI-Across-Industrial-Project-Delivery.pdf

Lee, K. (2026). Why AI code verification is becoming the new engineering bottleneck, *TechNode Global*, 15 September. https://technode.global/2026/09/15/enterprise-ai-code-verification-cognition-kevin-lee/

Leitão, F. N., Holanda, L. C. de and da Silva, A. G. (2024). Industrial goods cost estimating using machine learning techniques, *AACE International Transactions 2024*, DSAA-4395.

Love, P. E. D., Edwards, D. J. and Irani, Z. (2008). Forensic project management: An exploratory examination of the causal behavior of design-induced rework, *IEEE Transactions on Engineering Management*, 55(2), 234-247.

Miles, L. D. (1961). *Techniques of Value Analysis and Engineering*, New York: McGraw-Hill.

© 2026 Konstantin Lagutin

Mishra, N. (2025). 4 big challenges for project and engineering leaders to solve, Independent Project Analysis, 31 March. https://www.ipaglobal.com

Moynihan, M. C. and Allwood, J. M. (2014). Utilization of structural steel in buildings, *Proceedings of the Royal Society A*, 470(2168), 20140170. https://doi.org/10.1098/rspa.2014.0170

National Society of Professional Engineers (NSPE), Board of Ethical Review (2024). Artificial intelligence in engineering practice, Case No. 24-02.

North Carolina Board of Examiners for Engineers and Surveyors (NCBELS) (2025). Guidelines on the use of artificial intelligence in engineering and surveying services, 16 July. https://www.ncbels.org

Parasuraman, R. and Manzey, D. H. (2010). Complacency and bias in human use of automation: An attentional integration, *Human Factors*, 52(3), 381-410.

Patel, A. and Kumbharathi, V. (2026). False confidence: The risk of AI hallucinations, AACE International Conference & Expo 2026, DSAA-4843.

Professional Engineers and Geoscientists Newfoundland and Labrador (PEGNL) (2026). Practice guideline for the use of AI in professional work, May. https://pegnl.ca

Qodo (2026). The 2026 state of AI code quality report: Verification is the new bottleneck, 23 September. https://www.qodo.ai/blog/state-of-ai-code-quality-report-2026/

Reason, J. (2000). Human error: Models and management, *BMJ*, 320(7237), 768-770.

SAE International (2021). J3016: Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles, April.

Sadeh, H., Zhang, R., Gaedicke, C., Shahbodaghlou, F., Lee, M. J. and Todorov, D. (2026). Embracing generative AI in construction through a quantitative analysis and weighted score ranking of perceptions, applications, and complexities, *Journal of Construction Engineering and Management*, 152(5), 04026041. https://doi.org/10.1061/JCEMD4.COENG-17444

SAVE International (n.d.). About value methodology: The VM Job Plan. https://www.value-eng.org/page/AboutVM

Schäfer, P., Krinke, L. J., Wlotzka, M. and Asprion, N. (2026). Context is all you need: Toward autonomous model-based process design using agentic AI in flowsheet simulations, *AIChE Journal*, early access. https://doi.org/10.1002/aic.70488

Shellshear, E. (2026). Challenging overdesign at scale: The role of GenAI in capital projects, Boston Consulting Group, 3 March. https://www.bcg.com/x/the-multiplier/how-genai-tackles-overdesign-in-capital-projects

Simon, H. A. (1955). A behavioral model of rational choice, *Quarterly Journal of Economics*, 69(1), 99-118.

Technical University of Munich, Georg Nemetschek Institute (TUM GNI) (2026). Automated design compliance checking (ADCC) Global Summit 2026, Munich, 9 July. https://www.mdsi.tum.de/gni/events/

Unite.AI (2026). Autodesk previews agentic AI across Forma, Fusion and Flow, *Unite.AI*, September.

© 2026 Konstantin Lagutin

About the Author



Konstantin Lagutin

Israel

Konstantin Lagutin is an international PMO and capital-project management leader with 25+ years of experience across mining, metals, chemicals, oil and gas, and industrial transformation programs in the GCC, EMEA, Russia, CIS, Europe, and the United States. He holds an MBA and an MSc in Data and Business Analytics from Duke University in the U.S. and is a certified Project Management and MIT Digital Transformation professional.

Konstantin has built PMOs and project management systems from scratch for major Mining & Metals companies, supporting capital-project portfolios from $1 billion to $7.5 billion. His work spans stage-gate governance, digital PMOs, BIM-enabled and 4D construction management, and AI-supported value engineering. He is based in Israel and can be contacted at lagutin.ki@gmail.com | WhatsApp: +972 54 7567868.

© 2026 Konstantin Lagutin