quirq:
A Unit of Work for Intelligence

Suraj Sharma
XO Labs Inc
Correspondence: suraj@xo.builders

Draft v3, July 2026

Abstract

AI today is metered on one side only. The token is a genuinely good unit for what the machine consumes: it counts compute, scales with energy, and prices inference. But nothing counts what the machine *delivers*, and an input meter without an output meter cannot answer the only question a business asks: is this working? We propose the **quirq**, a unit of measurement for the business impact of agentic work. A quirq is minted, never self-reported: a human owner budgets an outcome at value B; an environment snapshots the world before and after execution and scores completion $V \in [0, 1]$ against a weighted definition of done; verification then mints $V \cdot B$ quirqs of delivered work. Against the minted quirqs we meter the *all-in* cost of production: inference tokens, CPU and GPU time, external API calls, storage, and the human interventions the work still required. From one ledger of units, every number an operator needs falls out: cost per quirq, quirq margin, the Quirq Efficiency Ratio (quirqs delivered per all-in dollar), quirq velocity, the intervention rate, and their trajectories over time, which quantify, week by week, how effective AI actually is inside a company. Tokens and quirqs are duals: one meters the machine's draw on the world, the other meters the world's change by the machine, and dividing them yields bridge metrics (quirqs per kilowatt-hour, quirqs per tonne CO_2) that connect AI's energy accounting to its economic justification. We develop the full calculus with worked arithmetic at every step, ground the budget denominator in a century of contract theory, confront the gaming attacks any unit of account invites, validate the machinery on an open harness with a content-addressed results ledger, and tier every claim (sourced, derived, measured, open) with a public validation program at https://docs.xo.builders.

1 Introduction: one meter is missing

Every consequential technology gets two meters. Electricity has the watt on the supply side and, on the demand side, the ledger of what the power actually produced: lumens, ton-miles, output per shift. Labor has the hour on the input side and the deliverable on the output side. An economy learns to use a technology exactly as fast as it learns to read both meters against each other.

AI currently has one. The **token** is a genuinely good input meter, better than its critics allow. It counts the atomic events of inference; it scales nearly linearly with floating-point operations and therefore with energy drawn and carbon emitted [14, 16]; it prices capacity in a way infrastructure providers can plan against. When the question is *what did the machine consume*, the token answers well, and Section 2 formalizes exactly how far that answer reaches.

But no business runs on an electricity bill. The question a CFO asks is not "how many tokens did we burn" but "what did we get, what did it cost all-in, and is the ratio improving." Today that question has no unit to be answered in. Enterprise AI budgets climb while returns stay undemonstrable [7], and a large fraction of generative-AI pilots show no measurable bottom-line effect [15], not because the systems do nothing but because a token bill is incommensurable with anything a business already measures. It swings with model choice, verbosity, and retries; it pools unrelated work into one line item; and it prices the machinery rather than any result. The measurement crisis of enterprise AI is not a data problem. It is a *units* problem.

This paper proposes the missing meter. The **quirq**[1] is a unit of measurement for delivered, verified, human-valued work. Its construction takes one paragraph and the rest of the paper makes it exact:

[1] From *quantum of irreducible work*. We spell it with a q at both ends because a quirq begins and ends in the same place: a check against the state of the world.

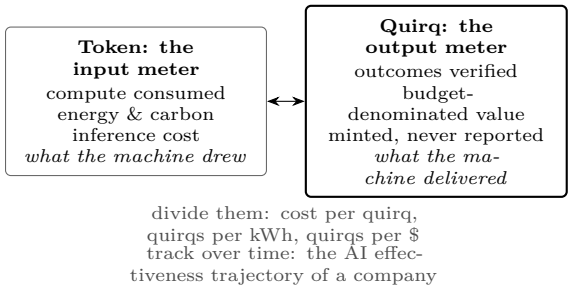

Figure 1: Two meters. The token is the right unit for AI's energy and compute impact; the quirq is the proposed unit for its business impact. Every metric in this paper is one of the two, or a ratio of them.

A human owner defines an outcome with a machine-checkable definition of done and budgets it at value B: the price they would pay for the outcome to exist. The task then carries B *potential quirqs*. An environment snapshots the world before execution and after, scores completion $V \in [0, 1]$ against the definition of done, and *mints* $V \cdot B$ quirqs of delivered work. Quirqs are never self-reported: they are minted by verification against captured state, and recorded in a tamper-evident ledger together with everything their production consumed.

Three properties make this a unit of account rather than another dashboard metric. **Human-denominated:** the budget B is set by the party who wants the outcome and pays for it, so a quirq is denominated in demonstrated willingness to pay, the only value signal an economy ultimately trusts. **Machine-minted:** verification is a state comparison computed by the environment that hosted the work, never the worker's own account, so the unit resists the self-report inflation that destroys activity metrics. **Cost-complete:** against minted quirqs the environment meters everything production consumed: inference tokens, CPU and GPU seconds, external API calls, storage, and the human minutes spent intervening when checks failed. The ratio of the two sides, quirqs delivered per all-in dollar, is the number that has been missing.

The token and the quirq are duals, and the pairing is the point (Figure 1). Tokens meter the machine's draw on the world: compute, energy, carbon, cost. Quirqs meter the world's change by the machine: outcomes delivered at human-assigned value. Neither replaces the other. Divided, they produce the bridge metrics this paper develops: cost per quirq, quirqs per kilowatt-hour, quirqs per tonne of CO_2, and, tracked over time inside one company, a trajectory that answers "is AI working here" with a trend line instead of an anecdote.

The paper proceeds as the construction requires. Section 2 states precisely what the token measures well and where it stops. Section 3 argues, from a century of contract theory, that the human-assigned budget is the only denominator that survives optimization pressure, and defines the unit of work whose lifecycle mints quirqs. Section 4 is the core of the paper: the complete quirq calculus, from the scoring rule through the all-in cost model to the portfolio and time-series metrics, with worked arithmetic at every step. Section 5 assembles the calculus into the company-level accounting it exists for: a month-over-month quirq ledger and the adoption path that produces it. Section 6 confronts the attacks any unit of account invites, budget inflation first. Section 7 is the validation program, hypothesis-first: every empirical claim stated with its falsifier and bound to numbered experiments, completed (E1–E3) and scheduled (E4–E7), on an open harness with the evidence honestly tiered. Section 8 compares the quirq against every unit software has tried before, and Section 9 tiers every claim this paper makes.

Our epistemic posture is unchanged from earlier drafts and applied to a bolder thesis: the hypotheses are stated at full strength, each carries a falsifier, the measured tier is backed by a re-runnable harness whose results ledger is content-addressed, and where evidence is currently mechanical rather than empirical, the caption says so, not a footnote.

2 What the token measures, exactly

Give the token its due before assigning its limits. Inference cost is, to first order, proportional to tokens processed: for a model with P active parameters, a forward pass costs roughly $2P$ floating-point operations per token, so a workload of N tokens implies $\approx 2PN$ FLOPs, which hardware converts to energy at its achieved efficiency η (FLOPs per joule):

$$E \approx \frac{2PN}{\eta}, \qquad \mathrm{CO}_2 \approx E \cdot c_{\mathrm{grid}}, \tag{1}$$

with c_{grid} the carbon intensity of the supplying grid. The constants move with architecture (mixture-of-experts activates a fraction of P), batching, caching, and datacenter overhead (multiply by PUE), and measured deployments show wide variance [14, 16]; but the *structure* is right, and it makes the token the natural unit of AI's physical footprint. An operator who knows their token volume, fleet efficiency, and grid can estimate energy and carbon from the same meter that prices their API bill. Token accounting is real accounting.

What the token cannot do is cross the boundary from cost to value, and the failure is structural, not a matter of better dashboards. Three independent breaks: **(i) Non-monotonicity.** More tokens do not mean more work: a verbose failure costs more than a terse success, and a retry loop costs most of all. Any metric that *rises*

when the work goes badly cannot denominate the work. **(ii) Model-relativity.** The same outcome costs 10× different token counts across models and prompts; a unit of account that changes size when the machinery is swapped is a ruler made of rubber. **(iii) Value-blindness.** Tokens spent resolving a $4 support ticket and tokens spent reviewing a $25,000 contract are indistinguishable in the bill. The meter pools what the business most needs separated.

So the division of labor is clean: tokens meter the machine side completely (cost, energy, carbon), and something else must meter the work side. The question is what that something's *denominator* should be.

3 Why the budget is the right denominator

Candidate output units fail in instructive ways. Counting *tasks* treats a typo fix and a migration as equal. Counting *artifacts* (lines, pages, commits) rewards volume, the SLOC failure [2]. Counting *expert-judged complexity* (function points, story points) reintroduces the human bottleneck the agents were meant to remove, and story points do not even transfer between teams. Letting the *worker* value its own output is self-report, the one design this paper exists to rule out.

The remaining candidate is the one the economy already uses everywhere else: the price the buyer commits to before the work starts. A century of contract theory says this is not a compromise but the load-bearing choice. Organizations exist because outcomes are cheaper to buy than to specify act-by-act [6, 18]; the essence of organization is metering contribution [3]; when effort is unobservable, contracts must be written on verifiable signals [11]; and when only some dimensions are measured, effort migrates to the measured ones [12], which is why the measured dimension had better be the outcome itself, valued by the party who pays for it. Hours became the unit of knowledge work precisely because outcomes were too expensive to verify; the agentic workspace, which observes every action as a side effect of hosting the work, removes that excuse. The budget B, set by the outcome's owner, is the hardest-to-game value signal available: it is a willingness to pay, committed before execution, by the only party with standing to say what the outcome is worth.

3.1 The unit of work: the contract that carries the budget

A budget needs a vehicle: something ownable, checkable, and settleable. Following the unit-of-work research program [19], that vehicle has three properties: a *definition of done* stated before execution, a *verifiable result* checked by state comparison on return, and a *single owner* accountable for acceptance. Its lifecycle is fixed: **define** the outcome and its checks; **budget** what reaching it is worth; **execute** inside a workspace; **verify** by comparing captured before- and after-state; **settle** the budget against metered cost. A prompt can carry none of this: it is stateless, holds no files, no tools, no budget, and no record, so whatever it produces is unowned and uncheckable. The unit of work lives instead in an *environment*: a workspace with a runtime, memory, files, tools, a budget, and a record.

The environment is not plumbing; it is the institution that makes the quirq mintable. It captures S_0 when the unit is created and S_1 when the agent reports done, so the score is computed on evidence the worker never produces. It meters every token, compute-second, and API call as a side effect of hosting the work, so the cost side of the ledger is complete by construction. And it writes the record as a hash chain, so history is tamper-evident by recomputation. Agents optimize what their environment measures, not what their principals intend [4, 10]; the environment is therefore the effective specification, the scorekeeper, and the binding constraint, and everything the quirq claims as integrity is inherited from it. Section 7 demonstrates each inheritance mechanically.

3.2 The mint

With the vehicle in place, the unit of measurement follows:

Quirq (mint rule). A unit of work u with owner-assigned budget $B(u)$ carries $B(u)$ potential quirqs. Upon verification at score $V(u) \in [0, 1]$ (Section 4), the environment mints

$$Q(u) \;=\; V(u) \cdot B(u)$$

quirqs of delivered work (divisible settlement), or $Q(u) = B(u) \cdot [V(u) \geq \tau]$ under atomic settlement, where the owner declared at creation that partial completion is worthless. Minted quirqs are appended to the ledger with the unit's full cost record.

Quirqs inherit the budget's currency: within a company, a quirq is one dollar (euro, rupee) of *verified, owner-valued, delivered* work, which makes quirq totals directly comparable to payroll, vendor spend, and revenue, the comparison the token bill could never support. Across companies and currencies, every metric this paper builds is a dimensionless ratio (quirqs per dollar of all-in cost, quirqs per kWh), so the unit travels. And the mint rule locates each concern where it can be governed: *what is it worth* is human judgment, priced once, before execution; *was it delivered* is machine verification, computed from state; *what did it cost* is metering, complete by construction. No component trusts the worker's testimony, and only the first trusts anyone at all.

4 The quirq calculus

This section is the paper's core: every calculation needed to run quirq accounting, with worked arithmetic. Nothing requires new infrastructure beyond an environment that snapshots state and meters spend.

4.1 Scoring: how completion is computed

Definition 1 (Unit of work)**.**
$u = (S_0, G, B)$, where S_0 is the snapshot of world state at creation (files, tickets, records: whatever the workspace observes); $G = \{(g_1, w_1), \ldots, (g_n, w_n)\}$ is the definition of done, each g_i a decidable predicate over a state snapshot with weight $w_i > 0$; and $B > 0$ is the owner's budget.

When the agent reports done, the environment captures S_1 and computes

$$V(u) = \frac{\sum_i w_i \, g_i(S_1)}{\sum_i w_i} \in [0,1], \quad \text{done}(u) = [V(u) \geq \tau]. \tag{2}$$

Every check evaluates S_1, captured by the environment, never the agent's account of what it did: the score is a property of the world, not of a report. *Worked:* a support ticket carries three checks, status closed (w=0.5), reply sent (w=0.3), knowledge-base article linked (w=0.2). The after-snapshot passes the first two: $V = (0.5 + 0.3 + 0)/1.0 = 0.8$. Under atomic settlement ($\tau = 1$) nothing mints until the third check passes; under divisible settlement the unit mints $0.8 \cdot B$ now.

4.2 The all-in cost of an outcome

The token bill is one line of the true cost. The environment meters all of them:

$$C_{\text{total}}(u) = \underbrace{\textstyle\sum_m N_m \, p_m}_{\text{inference}} + \underbrace{t_{\text{cpu}} r_{\text{cpu}} + t_{\text{gpu}} r_{\text{gpu}}}_{\text{compute}} + \underbrace{\textstyle\sum_j a_j \, p_j}_{\text{API calls}} + \underbrace{s \cdot r_{\text{store}}}_{\text{storage}} + \underbrace{\frac{F}{N_{\text{units}}}}_{\text{env. amort.}} + \underbrace{h \cdot r_{\text{human}}}_{\text{intervention}}, \tag{3}$$

where N_m is tokens on model m at price p_m; t are metered compute seconds at rates r; a_j counts calls to external service j at unit price p_j; s is storage occupied; F is the fixed cost of running the environment itself, amortized over the units it hosts; and h is human minutes spent on this unit when checks failed, at the intervenor's loaded rate. The last term matters most and is most often omitted: an AI program whose outputs each require twenty minutes of senior review is paying its largest cost in a currency the token bill never sees. Under quirq accounting it cannot hide, because interventions are exactly the $V < \tau$ events the scoring rule already counts.

Worked: the ticket above, resolved by an agent using 38,000 tokens at \$2/M on the primary model plus 4,000 at \$0.25/M on a cheap classifier: inference \$0.077. Sandbox time 90 CPU-seconds at \$0.04/hr: \$0.001. Two CRM API calls at \$0.01: \$0.02. Environment amortization \$0.03. No intervention. $C_{\text{total}} = \$0.128$.

4.3 Unit-level metrics

$$c_q(u) = \frac{C_{\text{total}}(u)}{Q(u)}, \qquad \mu(u) = Q(u) - C_{\text{total}}(u), \qquad x(u) = \frac{Q(u)}{C_{\text{total}}(u)}. \tag{4}$$

Cost per quirq c_q is the price of a dollar of verified work (dimensionless: dollars per quirq-dollar); quirq margin μ is the surplus on the unit; the multiple x is its reciprocal view. *Worked:* the ticket at $B = \$4.00$, $V = 1.0$ after the third check passes: $Q = 4.00$ quirqs, $c_q = 0.128/4.00 = 0.032$, margin \$3.87, multiple 31×. If instead the unit had settled divisibly at $V = 0.8$: $Q = 3.2$, $c_q = 0.040$. Incomplete work is automatically more expensive per quirq, which is the incentive pointing the right way.

4.4 Portfolio metrics: the company ledger

Over a set U of units in an accounting window T (a week, a sprint, a quarter):

$$\text{QER}(T) = \frac{\sum_{u \in U} Q(u)}{\sum_{u \in U} C_{\text{total}}(u)} \quad \text{quirq efficiency ratio,} \tag{5}$$

$$\text{QV}(T) = \frac{\sum_{u \in U} Q(u)}{|T|} \quad \text{quirq velocity,} \tag{6}$$

$$\text{IR}(T) = \frac{|\{u : V(u) < \tau\}|}{|U|} \quad \text{intervention rate.} \tag{7}$$

QER is the headline: quirqs of verified work delivered per all-in dollar. It is dimensionless, currency-independent, comparable across teams, vendors, and models, and it is what "AI ROI" should have meant all along. Velocity measures throughput in value terms rather than task counts, so it cannot be inflated by shipping confetti. The intervention rate is the trust signal: it is definitionally the failure share of Equation (2), measured for free, and because failures localize to named checks, it arrives with its diagnosis attached.

4.5 The time axis: quantifying effectiveness as a trajectory

A single QER reading is a snapshot; the thesis of quirq accounting is the *trajectory*. Two dynamics move it in opposite directions and must be separated.

Tenure: cost per quirq falls. Every unit's cost hides the cost of doing the work and the cost of discovering what the work is (reading the codebase, rediscovering

conventions, inferring what this owner means by done). Writing M_t for the environment's accumulated memory after t units,

$$C_{\text{total}}(u_t) = c_{\text{exec}} + k \cdot H(\text{intent} \mid M_t), \tag{8}$$

with c_{exec} the execution floor and H the residual intent uncertainty. In a persistent environment M_t is non-decreasing, so expected cost is non-increasing toward the floor; in a fresh context every unit pays cold-start cost. The identification of this decomposition on real agents is an open, testable claim (Section 7); its mock-mode consistency is demonstrated (85.6% decay to floor by unit six in the harness, versus a flat fresh-context arm).

Goodhart drift: measured value decays. Any proxy under optimization pressure degrades [10, 17]. For quirqs the pressure point is the check set: over time, agents find the shortest path to green, and the gap between checks-green and intent-satisfied widens unless checks are re-hardened. The observable is the *audit gap*

$$A(T) = \mathbb{E}\big[V_{\text{gold}}(u) - V(u)\big]_{u \in \text{audit}(T)}, \tag{9}$$

estimated on a random audit sample re-verified against gold checks held outside the production environment. Rising $|A|$ is the signal to re-specify before the ledger inflates.

The honest company-level effectiveness measure is therefore audit-corrected and trend-reported:

$$\text{QER}^*(T) = \text{QER}(T)\big(1 - A(T)\big), \quad \text{report } \tfrac{d\,\text{QER}^*}{dT},\ \tfrac{d\,\text{IR}}{dT}. \tag{10}$$

When QER^* rises while IR falls, AI is genuinely compounding inside the company: the same environment is delivering more verified value per all-in dollar with less human rescue. When QER rises while QER^* stalls, the checks are being farmed. When both stall while token volume grows, the company has bought an electricity bill.

4.6 The bridge metrics: joining the two meters

Because the cost model retains token counts, the physical accounting of Section 2 composes with the value accounting:

$$\frac{Q}{E} = \frac{\sum_u Q(u)}{\sum_u N(u)\, e_{\text{tok}}}\ [\text{quirqs/kWh}], \qquad \frac{Q}{\text{CO}_2}\ [\text{quirqs/tonne}], \tag{11}$$

with e_{tok} the fleet's measured energy per token and grid intensity converting to carbon. *Worked, illustrative parameters:* a month at 2.1B tokens with $e_{\text{tok}} = 1.5$ J/token implies $E \approx 875$ kWh; if the month minted 148,000 quirqs, the plant runs at ≈ 169 quirqs/kWh. These are the sustainability numbers a board can act on: not "we used less AI" but "we delivered more verified value per unit of energy," the same efficiency frontier every other industrial process is managed on.

Proposition 1 (Meter separation)**.**
Under the mint rule, token volume affects the quirq ledger only through C_{total}. No sequence of inference operations changes $Q(u)$ except by changing the world state that the checks evaluate.

This is the formal statement of the duality: the input meter and the output meter are coupled only through the world, which is exactly where a business wants them coupled.

5 The company dashboard: quirq accounting in practice

The calculus exists to produce one artifact: a ledger a company reads monthly the way it reads its P&L. Table 1 assembles a worked quarter for a hypothetical mid-size support-and-engineering operation running three unit types. Every number is computed from the equations of Section 4; the per-check pass probabilities and costs are drawn from the micro-benchmark ranges of the underlying research program [19]. The table is arithmetic, not measurement: its role is to exhibit the full calculation chain end to end, and the validation program's job is to replace it with production ledgers.

The reading discipline matters as much as the numbers. QER answers *is the program paying*; its audit-corrected trend (Equation (10)) answers *is it improving honestly.* The intervention rate answers *can it be trusted with more*, and its per-check decomposition names the next environment investment: in the worked quarter, one support-ticket check (knowledge-base linking) accounts for most interventions, so hardening that single check moves the whole company trajectory. Cost per quirq, split by unit type, prices each category of work against its human-baseline alternative, giving procurement an actual comparison: a contract review at $c_q = 0.13$ against a paralegal-hour baseline is a decision, not a vibe.

Adoption requires no big bang: **(1)** pick one recurring task family; **(2)** write its definition of done as weighted checks and let the owner budget it (the hardest step, and the one that was always implicit in delegation); **(3)** run it in an instrumented environment that snapshots, meters, and records; **(4)** read the ledger weekly; harden the worst check; repeat. The unit of account does the rest, because every equation in Section 4 is computed from data the environment already captured.

6 Gaming the quirq

A unit of account is a target, and targets get gamed [12, 17]. Quirq accounting does not escape Goodhart; it is engineered to fail loudly where activity metrics fail silently. The attack surface, in order of severity:

Budget inflation. If quirqs are the KPI, inflate B. Mitigations are structural: the budget is set by the party

Month	Units	Potential ($B)	Minted Q	Inference	Compute+API	Intervention	C_{total}	QER
April	2,100	18,400	15,770	$1,490	$410	$3,120 (312 h·$10)	$5,020	3.1×
May	3,400	29,900	26,310	$2,210	$630	$3,640	$6,480	4.1×
June	4,800	41,300	38,000	$2,730	$820	$3,280	$6,830	5.6×

Trend read: QER 3.1 → 5.6 (+81%); IR 18.1% → 11.4%; cost per quirq 0.32 → 0.18; quirq velocity +141%.
Same quarter in tokens alone: spend rose from $1,490 to $2,730 (+83%), a number indistinguishable from waste.

Table 1: A worked quarterly quirq ledger (illustrative arithmetic exhibiting the full calculation chain; parameters from the micro-benchmark ranges of XO Research [19]). The token bill, read alone, says costs nearly doubled. The quirq ledger says verified delivered value per all-in dollar rose 81% while human rescue fell a third: the difference between an expense line and an investment case.

paying it, so inflation is self-taxing wherever budgets clear against real money (outcome-priced vendors, internal chargebacks). Where budgets are notional, they must be benchmarked: against historical human cost for the same outcome, against market rates for comparable deliverables, and against post-hoc value audits on samples. The ledger makes inflation visible as a drifting ratio of budget to audited value; Equation (10)'s audit machinery covers value as well as completion. Residually: a company that lies to itself about what outcomes are worth had no unit of account before quirqs either; quirqs merely timestamp the lie.

Check farming. Manufacture units whose checks are trivially green. Countered by the same audit gap: gold-check sampling plus the owner's acceptance authority at settlement. A unit whose checks pass but whose owner rejects is an audit event, recorded, and the check set is re-specified.

Verification-surface attacks. Edit the test rather than fix the code. This is the one attack that is fully mechanical, and it is fully mechanically countered: the verification surface is itself under state comparison (byte-identical across the unit), and Experiment E2 shows a single such check converting 100% silent success into 100% detection.

Self-report injection. Convince the scorer to read the agent's summary. Ruled out by construction: the mint consumes only environment-captured state, and Experiment E1 quantifies exactly what re-admitting self-report costs (every false claim settles).

Salami slicing. Split one outcome into many units to harvest divisible partial credit. Countered by atomic settlement as the default for outcomes whose value is holistic, and by the owner's monopoly on unit creation: workers execute units; they do not define them.

The honest summary: quirq integrity reduces to environment integrity plus budget governance. The first is an engineering property, demonstrated below. The second is an institutional property, and Section 7.1 states it as an open hypothesis rather than an assumption, because that is what it is.

7 Validation: hypotheses and experiments

The validation program is organized hypothesis-first: each load-bearing empirical claim is stated at full strength with its falsifier, and each is bound to numbered experiments, completed (E1–E3) or scheduled (E4–E7). Evidence comes in two modes, never conflated: *mock mode* (scripted agents; validates the machinery, cannot validate claims about real agents) and *real mode* (identical protocol, production coding agent, measured tokens).

7.1 The hypotheses

Hypothesis 1 (Mint integrity)**.**
Completion read from environment-captured state mints no fiction: false completion claims settle at rate zero, while any self-report channel settles them at approximately the false-claim rate, and gameable check sets inflate the ledger until the verification surface itself is placed under state comparison. Falsifier: *the environment arm settling any incomplete unit, or hardened checks failing to detect verification-surface attacks.* Experiments: *E1, E2 (mock: complete, held); E4 (real mode: in progress).*

Hypothesis 2 (Ledger identification)**.**
On real production agents, the cost decomposition of Equation (8) identifies: persistent-environment cost per quirq falls with unit index toward a stable floor, fresh-context cost does not, holding model, prompt, and task family fixed. Falsifier: *flat or rising persistent-arm trajectories, or equal decay in fresh contexts.* Experiments: *E3 (mock consistency: complete, held); E4 (real mode: in progress).*

Hypothesis 3 (Predictive validity)**.**
Audit-corrected QER trend predicts real business outcomes of AI programs (renewal, expansion, P&L attribution) better than token spend, task counts, or benchmark scores. Falsifier: *in deployments instrumented with quirq ledgers, QER* trend fails to outperform those baselines as a predictor.* Experiments: *E5 (pilot ledgers), E6 (predictive study).*

Hypothesis 4 (Budget governance)**.**
Under the mitigations of Section 6 (payer-set budgets, benchmark anchoring, sampled value audits), budget inflation is bounded: the ratio of budgets to audited value stays within audit tolerance over sustained optimization pressure. Falsifier: *systematic budget drift in long-running ledgers despite the mitigations.* Experiments: *E7 (longitudinal budget-drift audit).*

7.2 Completed experiments (E1–E3, mock mode)

The machinery is implemented in an open reference harness[2] whose results ledger is content-addressed (`sha256:ce4d91c4...aa2154`), so a skeptic re-runs and compares hashes rather than trusting this paragraph.

E1, verification source (Hypothesis 1). 200 identical units, an agent that falsely claims done with probability 0.05. Self-report arm: all 7 false claims silently settle and mint (3.5% of the ledger is fiction). Environment arm: the same 7 claims, zero mint. The mint rule's core integrity property, exhibited.

E2, hardening (Hypothesis 1). 50 units, an agent that guts the test instead of fixing the code. Gameable definition of done: checks green 100%, gold-verified intent 0%: a maximally inflated ledger. Plus one verification-surface check: detection 100%. The audit gap of Equation (9), and its closure, in miniature.

E3, tenure (Hypothesis 2). Twelve similar units, persistent versus fresh environment: cost per unit decays 85.6% to the execution floor versus flat cold-start (Figure 2). Mock-mode caveat in full: the scripted agent implements Equation (8), so this arm demonstrates harness consistency, not agent behavior.

7.3 The experiment roadmap (E4–E7)

E4, real-mode replication (Hypotheses 1, 2): E1–E3 re-run with a production coding agent and measured tokens; the false-claim rate becomes a measurement rather than a parameter, and the tenure curve becomes evidence. In progress; its Figure 2 replacement is the program's current deliverable. **E5, pilot ledgers** (Hypothesis 3): the dashboard of Section 5 instrumented on real work across at least three unit types, replacing Table 1's arithmetic with production data. **E6, predictive study** (Hypothesis 3): across pilot deployments, QER* trend versus token spend, task counts, and benchmark scores as predictors of renewal, expansion, and P&L attribution. **E7, budget-drift audit** (Hypothesis 4): longitudinal ratio of budgets to sampled audited value in long-running ledgers under the Section 6 mitigations. Results and data are published as they land at `https://docs.xo.builders`.

[2]Dependency-free Python: snapshots, weighted checks, the mint and settlement rules, hash-chained ledger, mock agents, and a runner for a production coding agent. Pre-registered specs state hypothesis, method, metric, and falsifier before results. Code, specs, and per-run data: `https://docs.xo.builders`.

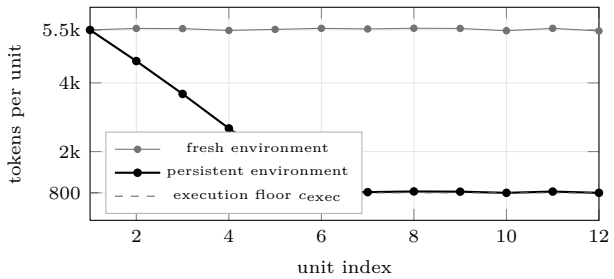

Figure 2: E3, harness output (mock mode): cost per unit across twelve similar units. Persistent memory decays 85.6% to the floor; fresh context pays cold start every time. The scripted agent implements Equation (8), so this is a consistency demonstration; real-mode measurement is in progress.

8 Related units of account

Software has priced work before, and each attempt teaches a constraint the quirq is built against. **SLOC** counts artifacts and rewards volume. **Function points** [2] price specified functionality, a real advance, but need expert human counters and measure specification size rather than delivered change; quirqs are machine-minted from state. **Story points** are deliberately relative and team-local, useless for pricing across organizations; quirqs inherit currency and travel as ratios. **DORA and SPACE** [8, 9] measure the delivery system and explicitly decline to price a worker's deliverable, a wise refusal for humans that leaves agents unpriced; quirqs price the deliverable while keeping attribution at the unit, not the person. **Execution-gated benchmarks** [13] share the quirq's verification-by-state core but score against researcher-authored tests with no budget semantics: benchmarks measure capability, quirqs measure delivered value. **Tokens**, finally, are the meter the quirq is dual to, not a competitor: Section 2 is their defense. In the task-model frame of labor economics [1, 5], quirq ledgers are, as a side effect, task-level automation data at market scale: which check types agents settle cheaply, and which still route to humans, is the substitution margin, observed rather than surveyed.

9 The claim ledger

10 Limitations

The budget is the model's entry point for human error: a mispriced outcome mints mispriced quirqs, and governance (Hypothesis 4) is institutional, not mechanical.

Claim	Tier
Tokens track compute, energy, carbon to first order	sourced
Token bills cannot denominate business value	sourced
Measured proxies distort effort; outcomes must be the measure	sourced
Mint rule: $Q = V \cdot B$; scoring, settlement, IR are one computation	derived
Meter separation: inference volume cannot mint quirqs	derived
Tenure: cost non-increasing under non-decreasing memory	derived
Self-report mints fiction; state-minting does not	measured (E1, mock)
One surface check: silent gaming to full detection	measured (E2, mock)
Cost decays to floor in persistent environments	measured (E3, mock; consistency only)
Dashboard arithmetic (Table 1)	illustrative
Real agents reproduce E1–E3 (Hyps. 1, 2; E4)	open
QER* predicts program outcomes (Hyp. 3; E5, E6)	open
Budget inflation is governable (Hyp. 4; E7)	open

Table 2: Every load-bearing claim, tiered. Open claims carry falsifiers in the text and validation commitments at https://docs.xo.builders.

Definitions of done are incomplete contracts; work whose acceptance is irreducibly judgmental enters the ledger only through the owner's verdict, which restores a human bottleneck exactly where the work is fuzziest. The energy bridge inherits the wide measured variance of per-token energy [14]. The measured tier is mock mode, validating machinery rather than agents. And a unit of account reshapes the behavior of what it measures; our mitigations are designed, exhibited in miniature, and unproven at scale, which is why the validation program, not this paper, is the product.

11 Conclusion

The token meters what AI consumes; nothing has metered what it delivers. The quirq is that meter: potential value budgeted by a human owner, minted by machine verification against captured world state, costed all-in from inference to intervention, and read over time as a trajectory. One ledger yields the cost of a dollar of verified work, the efficiency of an AI program, the trust signal for extending its autonomy, and the energy-to-value bridge a board can govern by. The construction is deliberately conservative: a scoring rule, a cost sum, and ratios, computable today by any environment that snapshots state and meters spend. The claims that matter are correspondingly empirical, and they are on the record with falsifiers attached. If the program succeeds, "is AI working here" becomes a number with a trend, and the agentic workforce gets what every workforce before it eventually got: an honest unit of account. If it fails, the ledger will say so, which is the point of having one.

References

[1] Daron Acemoglu and Pascual Restrepo. 2019. Automation and new tasks: How technology displaces and reinstates labor. *Journal of Economic Perspectives*, 33(2):3-30.

[2] Allan J. Albrecht. 1979. Measuring application development productivity. In *Proceedings of the Joint SHARE/GUIDE/IBM Application Development Symposium*, pages 83-92.

[3] Armen A. Alchian and Harold Demsetz. 1972. Production, information costs, and economic organization. *American Economic Review*, 62(5):777-795.

[4] Alignment Research. 2025. Broad emergent misalignment from exploitable production environments. *arXiv preprint arXiv:2511.18397*.

[5] David H. Autor. 2015. Why are there still so many jobs? The history and future of workplace automation. *Journal of Economic Perspectives*, 29(3):3-30.

[6] Ronald H. Coase. 1937. The nature of the firm. *Economica*, 4(16):386-405.

[7] Deloitte. 2024. State of AI in the enterprise. Technical report, Deloitte Insights.

[8] Nicole Forsgren, Jez Humble, and Gene Kim. 2018. *Accelerate: The Science of Lean Software and DevOps*. IT Revolution Press.

[9] Nicole Forsgren, Margaret-Anne Storey, Chandra Maddila, Thomas Zimmermann, Brian Houck, and Jenna Butler. 2021. The SPACE of developer productivity. *ACM Queue*, 19(1):20-48.

[10] Leo Gao, John Schulman, and Jacob Hilton. 2022. Scaling laws for reward model overoptimization. *arXiv preprint arXiv:2210.10760*.

[11] Bengt Holmström. 1979. Moral hazard and observability. *Bell Journal of Economics*, 10(1):74-91.

[12] Bengt Holmström and Paul Milgrom. 1991. Multitask principal-agent analyses: Incentive contracts, asset ownership, and job design. *Journal of Law, Economics, and Organization*, 7:24-52.

[13] Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. 2023. SWE-bench: Can language models resolve real-world GitHub issues? *arXiv preprint arXiv:2310.06770*.

[14] Alexandra Sasha Luccioni, Yacine Jernite, and Emma Strubell. 2023. Power hungry processing: Watts driving the cost of AI deployment? *arXiv preprint arXiv:2311.16863*.

[15] MIT / The AI Consulting Network. 2025. Enterprise AI ROI: Why most generative-AI pilots show no measurable P&L impact. Industry report.

[16] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon emissions and large neural network training. *arXiv preprint arXiv:2104.10350*.

[17] Marilyn Strathern. 1997. 'Improving ratings': Audit in the British university system. *European Review*, 5(3):305-321.

[18] Oliver E. Williamson. 1975. *Markets and Hierarchies: Analysis and Antitrust Implications*. Free Press.

[19] XO Research. 2026. Why the unit of work: A research note. https://docs.xo.builders/future-of-work/phase-1-agentic-workforce/unit-of-work-research.