MONTREAL.AI · QUEBEC.AI · RESEARCH PREPRINT · CEILING EDITION

Forecasting a World That Accelerates

An Evidence-Grounded, Constraint-Limited, Regime-Switching Framework for AI-Era Decisions

Compact canonical model

$$Y_s(t) = B_s(t)\min\left\{Q_0\exp\left[\int_0^t g_{Q,s}(u)\,\mathrm{d}u\right],\ M_s(t)D_s(t)U_s(t)\right\}$$

Realized output is the lesser of what can be technically produced and what the relevant market, institution, or mission can absorb, after residual bottlenecks are applied.

Vincent Boucher
President, MONTREAL.AI & QUEBEC.AI
Montréal, Québec, Canada · info@quebec.ai

Version 2.0.0 · Ceiling Edition · July 28, 2026 · Research cutoff: July 28, 2026 (America/Montréal)
Canonical repository: https://github.com/MontrealAI/forecasting-a-world-that-accelerates

Abstract

Forecasts of AI-linked outcomes commonly extrapolate benchmark performance, compute, price, adoption, or economic output as though they were interchangeable. That shortcut is structurally fragile: capability is not reliable execution; reliable execution is not verified completion; verified completion is not authorized deployment; deployment is not adoption; and adoption is not realized economic or mission output. The problem becomes more severe when capability, cost, task duration, automation, parallelism, reliability, demand, and institutional constraints evolve at different rates, and when the law governing those rates may itself change during the forecast horizon. We introduce an acceleration-aware, constraint-limited, regime-switching framework for forecasting realized outcomes. Its compact canonical model defines realized output as the residual-bottleneck-adjusted minimum of technical capacity and absorptive demand. A generalized realization family then separates technical, demand, and transfer bottlenecks and permits partial substitution while retaining the hard-min model as an interpretable limiting case. Technical-output growth is decomposed into capability, cost decline, task-time compression, automation, effective parallelism, and verified reliability. A task-graph layer maps model-level progress into retries, verification, human fallback, authority latency, critical paths, correlated failure, and cost per verified outcome. The reference engine fits competing regimes, generates residual-bootstrap model-averaged predictive distributions, applies downside, accelerated, and hazard-driven discontinuity transformations, calculates milestone probabilities, renders a self-contained report, and can append the frozen forecast to a separate hash-chained registry after an explicit registration step. The release supplies a complete natural-language protocol, strict machine-readable contracts, deterministic controlled experiments, source-linked METR and SWE-bench demonstrations, an operational-evidence panel, a prospective validation design, and reproducible release infrastructure. The paper does not claim that each constituent equation is novel or that the demonstrations prospectively validate future AI trajectories. Its contribution is an integrated, auditable, falsifiable bridge from fresh evidence to verified, authorized, and realized outcomes in a world whose rate of change may itself change.

Research contribution. The framework makes the following chain explicit and testable:

$$\text{capability} \neq \text{reliable execution} \neq \text{verified completion},$$

$$\text{authorized deployment} \neq \text{adoption} \neq \text{realized value},$$

$$\text{forecast} \rightarrow \text{dated evidence} \rightarrow \text{probabilistic scenarios} \rightarrow \text{action} \rightarrow \text{prospective score}.$$

It is designed to prevent two symmetric failures: converting technical gains directly into deployment or revenue, and preserving historical execution timelines after automation, parallelism, verification cost, or authority latency have materially changed.

Keywords: technological forecasting; artificial intelligence; endogenous acceleration; regime switching; task graphs; reliability; adoption; bottlenecks; probabilistic forecasting; hindcasting; prospective validation; decision-making under deep uncertainty.

Status and disclosure. Version 2.0.0 is a versioned research preprint, executable protocol, and reference forecasting engine. It is not an audit, certification, assurance engagement, investment recommendation, legal opinion, or promise of forecast accuracy. The principal author directed the research design, assumptions, editorial decisions, and release. Generative AI systems assisted with research, drafting, coding, testing, and document production. The author remains responsible for the published content. No DOI, public repository state, or external preprint identifier is claimed until one is actually issued.

Contents

1 Introduction

Forecasting during gradual change is difficult. Forecasting while the rate of change is itself changing is categorically harder. A timeline inferred from yesterday's capability, cost, or execution speed can be obsolete before the forecast horizon ends. Yet simply replacing a conservative extrapolation with an aggressive exponential is not enough. Technical progress does not automatically become dependable work; dependable work does not automatically receive authority; authorized systems do not automatically diffuse through institutions; and adopted tools do not automatically remove physical, legal, capital, energy, supply-chain, or demand constraints.

Recent empirical work makes each part of this separation increasingly salient. Frontier-agent task horizons have advanced rapidly but remain task-suite-specific and uncertain outside their measurement domain (Kwa et al., 2025; METR, 2026). Cost-adjusted capability can improve faster than raw benchmark scores reveal (Gundlach et al., 2025; Erol et al., 2025), while agent reliability can lag capability gains (Rabanser et al., 2026). Production evidence in software shows that large gains in coding activity attenuate between commits, projects, releases, and actual use (Demirer et al., 2026). Economic models likewise emphasize weak links, automation shares, adjustment costs, diffusion, and resource constraints (Erdil et al., 2025; Jones, 2025, 2026; Trammell and Korinek, 2023; Davidson et al., 2026).

The resulting forecasting problem is not one-dimensional. It requires an evidence clock, a target definition, a translation from capability to verified work, a demand/absorption branch, a bottleneck operator, competing dynamic regimes, uncertainty over regime transitions, and an action rule that remains sensible when probabilities are unstable.

1.1 Research question

This paper asks:

> How should realized outcomes be forecast when capability, cost, task duration, automation, parallelism, reliability, adoption, and constraints evolve at different rates—and when the law governing those rates may change during the forecast itself?

The answer is a modular framework rather than a universal point forecast. Its compact core is

$$Y_s(t) = B_s(t) \min\left\{ Q_0 \exp\left[\int_0^t g_{Q,s}(u)\, \mathrm{d}u \right],\ M_s(t) D_s(t) U_s(t) \right\}. \tag{1}$$

The first branch is technical capacity; the second is absorptive demand; $B_s(t)$ represents residual constraints not already counted in either branch. Scenario s permits the governing dynamics to differ across base, accelerated, and discontinuous regimes.

1.2 Contributions

Version 2.0.0 makes eight integrated contributions.

1. **Realized-outcome architecture.** It models realized output as the bottleneck-adjusted minimum of technical capacity and absorptive demand rather than as a direct function of capability alone.
2. **Generalized realization family.** It embeds the hard minimum in a branch-specific constant-elasticity family so substitutability and transfer constraints are explicit rather than hidden.
3. **Acceleration accounting.** It separates metric levels, proportional growth rates, acceleration of those rates, bounded-variable transformations, and explicit 6–12 month comparisons.

4. **Capability-to-work translation.** It maps task success probabilities into retries, verification, fallback, authority latency, critical paths, parallel efficiency, correlated failure, and cost per verified outcome.
5. **Regime uncertainty.** It compares linear, exponential, accelerating, decaying-acceleration, constrained, change-point, and recursive-improvement dynamics instead of assuming a single trajectory.
6. **Forecast-to-decision layer.** It selects no-regret and option-preserving actions using expected value, learning value, option value, downside risk, and minimax regret.
7. **Probabilistic executable specification.** It publishes the full prompt, strict schemas, complete engine, residual-bootstrap model averaging, milestone distributions, self-contained reports, tests, and release machinery as a coordinated research artifact.
8. **Prospective validation contract.** It provides a hash-chained registry, preregistration template, proper scoring code, independent-replication procedure, and explicit gates separating release quality from validated predictive performance.

The novelty claim is deliberately bounded. Linear growth, exponential growth, logistic diffusion, item-response models, survival hazards, critical-path analysis, information criteria, model averaging, and conditional value-at-risk all have established antecedents. The contribution lies in their integration and operationalization for acceleration-aware forecasts of realized outcomes.

1.3 Design principles

The framework is governed by seven principles:

1. use the newest comparable evidence and record its exact clock;
2. never silently equate benchmark performance with economic realization;
3. model verification, authority, and adoption as first-class variables;
4. recalculate work from the current task graph rather than inherited timelines;
5. fit multiple regimes and expose model uncertainty;
6. treat a recursive-improvement transition as conditional and constrained, not decorative;
7. end with a dated monitoring and action policy, not only a projection.

2 Related Work and Positioning

2.1 Technological forecasting and model uncertainty

Classical forecasting distinguishes trend, seasonality, structural breaks, diffusion, and uncertainty, while emphasizing evaluation against held-out observations (Hyndman and Athanasopoulos, 2021; Hamilton, 1989; Diebold and Mariano, 1995). Information-theoretic selection and multimodel inference provide a principled alternative to choosing one apparently best in-sample curve (Akaike, 1974; Burnham and Anderson, 2002). Proper scoring rules evaluate probabilistic forecasts rather than rewarding overconfident point estimates (Gneiting and Raftery, 2007). The present framework adopts these foundations but adds a target-specific translation from technical progress to realized output and an explicit recursive-improvement regime.

2.2 AI capability, cost, and reliability measurement

Task-completion time horizons translate agent performance into the human-expert duration of tasks at a specified success probability (Kwa et al., 2025). METR's current public page reports an expanded suite and warns that measurements above 16 hours are unreliable with that suite (METR, 2026). This metric is valuable but is neither runtime nor a general measure of economy-wide automation. Cost-of-pass and price-performance work shows why a benchmark score without the cost of obtaining a correct result can misstate deployable progress (Erol et al., 2025; Gundlach et al., 2025). Reliability research further shows that consistency, robustness, predictability, and bounded failure severity cannot be compressed into one benchmark success rate (Rabanser et al., 2026).

2.3 Automation, diffusion, and economic bottlenecks

GATE integrates compute-based AI development, automation, investment, adjustment costs, and semi-endogenous growth (Erdil et al., 2025). Work on AI in R&D emphasizes the share of research tasks automated, productivity on those tasks, and bottleneck strength (Jones, 2025). Weak-link and O-ring mechanisms explain why a chain of complementary tasks can be limited by its least substitutable component (Kremer, 1993; Jones, 2026). Empirical evidence on coding tools demonstrates a production hierarchy in which activity gains attenuate before releases and use (Demirer et al., 2026). Bass diffusion and related adoption models supply useful demand-side dynamics (Bass, 1969).

2.4 Endogenous acceleration and recursive improvement

Economic models have long considered R&D-driven growth and possible departures from stable growth rates (Jones, 1995; Nordhaus, 2021; Trammell and Korinek, 2023). Recent work derives conditions under which automation across innovation networks can overcome diminishing returns and generate superexponential growth (Davidson et al., 2026). A 2026 survey distinguishes bounded self-refinement from increasingly closed research loops and stresses the central role of evaluators, grounding, compute, and human direction-setting (Chen et al., 2026). The discontinuity model below is therefore not an assertion that a singularity will occur. It is a conditional stress model whose parameters must be tied to independently verified loop closure and whose output remains constrained by physical and institutional capacity.

2.5 Forecasting systems and evaluation

SWE-bench supplies a reproducible real-world software issue benchmark, while its Verified subset focuses on human-screened solvable tasks (Jimenez et al., 2024; SWE-bench, 2026). GitHub operational statistics further illustrate that adoption volume and workflow latency require their own denominators rather than serving as direct causal productivity estimates (GitHub, 2025, 2026). ForecastBench demonstrates the value of dynamic, future-resolving questions and leakage-resistant evaluation for AI forecasting systems (Karger et al., 2024). A recent review of AGI forecasting methods calls for better data, repeated updating, adversarial validation, and explicit treatment of methodological gaps (Sarma et al., 2026). The present framework complements event-probability forecasting by concentrating on quantitative realized outcomes, task-graph capacity, and regime-dependent decision policies.

2.6 Positioning

No claim is made that the literature lacks adjacent integrated models. Rather, the paper's testable proposition is that forecast quality and decision usefulness improve when the following chain is made explicit in one protocol:

$$\begin{aligned}\text{fresh evidence} &\to \text{pace and acceleration} \to \text{technical capacity}\\ &\to \text{verified execution} \to \text{authorized deployment}\\ &\to \text{adoption} \to \text{realized outcome}\\ &\to \text{action and recalculation.}\end{aligned}$$

That proposition can be evaluated by hindcasts, ablations, calibration, decision regret, and independent replication.

3 The Evidence Clock and Pace Measurement

3.1 Target before trajectory

A forecast begins by defining the dependent variable $Y(t)$. The unit, scope, baseline date, baseline value, success criterion, and realization conditions must be explicit. A revenue forecast, for example, is not interchangeable with an adoption forecast; a completion-date forecast is not interchangeable with a capability threshold; a bounded probability should not be fitted as an unbounded positive quantity.

Each material input is labelled as *observed*, *derived*, *estimated*, *assumed*, or *scenario-conditional*. Vague predicates such as "autonomous," "commercial," or "complete" are converted into measurable acceptance criteria. If the target cannot be observed directly, the analyst publishes the proxy construction and weights rather than hiding them in prose.

3.2 Four dates, not one

A current-looking source can contain stale measurements. The evidence ledger therefore separates:

1. event date;
2. measurement or benchmark-execution date;
3. model or product release date;
4. publication or source-update date.

The research cutoff is an exact timestamp and time zone. Sources receive a grade: A for primary/directly reproducible, B for strong independent transparent analysis, C for indirect or self-reported evidence, and D for speculative or promotional material. A base forecast should not rest principally on C/D evidence when stronger evidence exists.

3.3 Comparable rates and acceleration

For a positive metric $X(t)$ and a window h measured in years, the continuously compounded rate is

$$g_X^{(h)}(t) = \frac{\ln X(t) - \ln X(t-h)}{h}. \tag{2}$$

Recent and prior windows yield

$$a_X(t) = \frac{g_X^{(h)}(t) - g_X^{(h)}(t-h)}{h}. \tag{3}$$

The equivalent annual proportional improvement and doubling time are

$$G_X = e^{g_X} - 1, \qquad T_2 = \frac{\ln 2}{g_X} \quad (g_X > 0). \tag{4}$$

For lower-is-better variables, the relevant series is inverted:

$$E_c(t) = \frac{c_0}{c(t)}, \qquad S_\tau(t) = \frac{\tau_0}{\tau(t)}. \tag{5}$$

For bounded variables such as adoption, automation, or success probability, log-odds changes can be more stable near the boundaries:

$$\operatorname{logit}(x) = \ln \frac{x}{1-x}. \tag{6}$$

Comparability rule. A benchmark revision, changed task distribution, altered scaffold, pricing-unit change, context-window change, or revised scoring rule creates a new measurement regime unless an explicit bridge is constructed. Two successive releases improving does not by itself establish acceleration.

3.4 Required pace matrix

The operational protocol requests current, roughly six-month-prior, and roughly twelve-month-prior observations for capability, cost per attempt, cost per verified outcome, task time, human supervision, automation share, autonomous horizon, parallelism, verified reliability, adoption, utilization, demand, compute, energy, capital, authority, trust, regulation, supply chains, and physical execution where relevant. The purpose is not to multiply every trend. It is to identify which factors are independent drivers, which are correlated manifestations of the same advance, and which have become binding constraints.

4 Compact Canonical Model and Generalized Realization Family

4.1 The compact canonical model

Let $s \in \mathcal{S} = \{\text{downside, base, accelerated, discontinuous}\}$ denote a scenario. Define technical capacity

$$Q_s(t) = Q_0 \exp\left[\int_0^t g_{Q,s}(u)\,\mathrm{d}u\right] \tag{7}$$

and absorptive demand

$$Z_s(t) = M_s(t) D_s(t) U_s(t). \tag{8}$$

The compact canonical model is

$$\boxed{Y_s(t) = B_s(t) \min\left\{Q_0 \exp\left[\int_0^t g_{Q,s}(u)\,\mathrm{d}u\right],\ M_s(t) D_s(t) U_s(t)\right\}.} \tag{9}$$

Definitions are:

- $Y_s(t)$: realized target outcome under scenario s;
- Q_0: baseline technically deliverable output;
- $g_{Q,s}(t)$: technical-output growth rate;
- $M_s(t)$: addressable market, mission volume, or demand pool;
- $D_s(t) \in [0, 1]$: eligible adoption or penetration fraction;
- $U_s(t)$: utilization or realized value per adopter;
- $B_s(t) \in (0, 1]$: residual realization bottleneck not already represented elsewhere.

The min operator is substantive. Output cannot exceed the amount that can be technically supplied or the amount the relevant institution, market, or mission can absorb. A system can be highly capable but unused; demand can be abundant while production remains technically constrained. The model is deliberately memorable and conservative about complementarity.

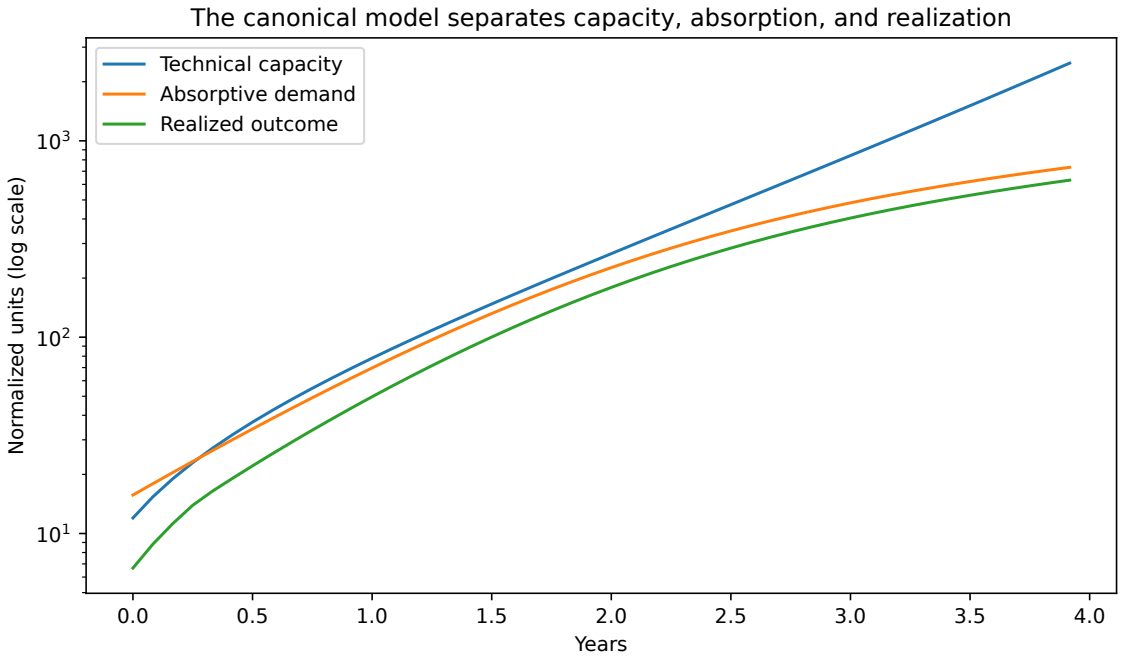

Figure 1: Illustrative compact canonical branches. Realized output follows the lower branch and is then reduced by constraints not already represented in that branch. The figure is controlled, not an empirical forecast.

4.2 Generalized realization family

The hard minimum is not the only defensible aggregation law. Some domains permit partial substitution, and constraints can operate asymmetrically on technical supply, absorptive demand, or the transfer between them. Version 2.0.0 therefore embeds Equation (9) in a generalized constant-elasticity realization family:

$$\boxed{Y_s(t) = B_s^X(t)\left[\alpha_s\left(B_s^Q(t)Q_s(t)\right)^{-\rho_s} + (1-\alpha_s)\left(B_s^Z(t)Z_s(t)\right)^{-\rho_s}\right]^{-1/\rho_s}.} \tag{10}$$

Here B_s^Q acts on technical production, B_s^Z acts on absorptive demand, B_s^X acts on realization or transfer, $\alpha_s \in (0, 1)$ controls branch weight, and $\rho_s > 0$ controls substitutability. As $\rho_s \to \infty$,

$$Y_s(t) \to B_s^X(t)\min\{B_s^Q(t)Q_s(t), B_s^Z(t)Z_s(t)\}. \tag{11}$$

Equation (9) is recovered when the branch-specific factors equal one and $B_s^X = B_s$. Finite ρ should be used only when substitution has domain evidence; otherwise the hard minimum is the transparent baseline. The reference engine supports both and requires the operator choice to be recorded.

4.3 Technical-output decomposition

The technical-output growth rate is

$$\boxed{g_{Q,s}(t) = \beta_\theta \dot{\theta}_s(t) - \beta_c \frac{\dot{c}_s(t)}{c_s(t)} - \beta_\tau \frac{\dot{\tau}_s(t)}{\tau_s(t)} + \beta_A \frac{\dot{A}_s(t)}{A_s(t)+\varepsilon} + \beta_P \frac{\dot{P}_s(t)}{P_s(t)} + \beta_R \frac{\dot{R}_s(t)}{R_s(t)}.} \tag{12}$$

Here θ is latent capability, c cost per attempt or unit of work, τ task-completion time, A automatable workflow share, P effective parallelism, R verified reliability, β_j target-specific elasticities, and $\varepsilon > 0$ a stabilizer.

Equation (12) is an accounting and estimation structure, not permission to set every elasticity to one. Capability, cost, speed, automation, parallelism, and reliability often share causes. Their coefficients are not causally identified by a short aggregate time series. A defensible implementation must estimate them from panel or task-level variation, constrain them using domain knowledge, or declare them scenario assumptions and perform sensitivity analysis.

4.4 Capability measurement

Raw benchmark averages saturate and change composition. A latent item-response representation is

$$\Pr(\text{success on item } i \mid t) = \sigma[a_i(\theta(t) - b_i)], \qquad \sigma(x) = \frac{1}{1+e^{-x}}, \tag{13}$$

where b_i is item difficulty and a_i discrimination (Rasch, 1980). For an economically meaningful task, capability becomes deployable only after its success probability, verification burden, failure severity, authority, and cost jointly satisfy the acceptance rule.

4.5 Residual bottlenecks

For constraint factors $b_k(t) \in (0, 1]$ and weights $w_k \geq 0$ summing to one, a smooth weakest-link aggregator is

$$B_\varrho(t) = \left[\sum_{k=1}^{K} w_k b_k(t)^{-\varrho}\right]^{-1/\varrho}, \qquad \varrho > 0. \tag{14}$$

As $\varrho \to \infty$, B_ϱ approaches the minimum factor. Smaller ϱ permits several constraints to matter jointly. The notation ϱ is distinct from the realization-family parameter ρ_s.

4.6 Double-counting and dimensional audit

Every material factor is assigned to exactly one substantive location: technical capacity, absorptive demand, branch-specific bottleneck, or transfer bottleneck. It may additionally appear as a diagnostic but not as a second penalty. If legal authorization is already embedded in eligible adoption $D(t)$, applying the same eligibility

fraction again in B^Z is double counting. If workflow reliability already determines cost per verified output, multiplying by that identical reliability again may be inappropriate.

Before fitting, the analyst must also verify dimensional consistency. Q_s and Z_s must be expressed in compatible units and over the same time basis. A stock cannot be minimized directly against a flow without conversion; a probability cannot be multiplied into currency without a defined exposure or quantity; a benchmark percentage cannot be interpreted as mission throughput without a mapping model.

Plain-language interpretation. Realized output is governed by what can be produced, what can be absorbed, and what can cross the boundary between them. The compact model uses a hard weakest link. The generalized model permits explicit, evidence-backed asymmetry and partial substitution without hiding those choices.

5 Candidate Growth Regimes and Discontinuities

No model class is assumed ex ante. The protocol fits or evaluates a family whose members encode different causal stories.

5.1 Linear and exponential baselines

$$X_L(t) = X_0 + vt, \tag{15}$$

$$X_E(t) = X_0 e^{gt}. \tag{16}$$

The linear model is useful as a conservative absolute-change baseline. The exponential model assumes a stable proportional rate.

5.2 Acceleration and decaying acceleration

If the proportional growth rate rises linearly, $g(t) = g_0 + at$, then

$$X_A(t) = X_0 \exp\left(g_0 t + \tfrac{1}{2} a t^2\right). \tag{17}$$

Constant acceleration is often too aggressive. If acceleration decays as $\dot{g}(t) = a_0 e^{-\kappa t}$, then

$$X(t) = X_0 \exp\left[g_0 t + \frac{a_0}{\kappa} t - \frac{a_0}{\kappa^2}\left(1 - e^{-\kappa t}\right)\right]. \tag{18}$$

This model permits current acceleration while making its persistence an estimable parameter.

5.3 Constrained and moving-ceiling models

A generalized logistic process is

$$\dot{X} = rX\left[1 - \left(\frac{X}{K}\right)^{\nu}\right]. \tag{19}$$

The ceiling K need not be a permanent physical limit. It can represent capacity under the current compute, regulation, capital, demand, or institutional regime. A moving-ceiling model combines acceleration and constraints:

$$\dot{X} = g(t) X B(t)\left[1 - \left(\frac{X}{K(t)}\right)^{\nu}\right]. \tag{20}$$

5.4 Change points and regime switching

A continuous one-change log-linear model is

$$\ln X(t) = \alpha + g_1 t + \Delta g\,(t-\tau)_+ + \epsilon_t, \tag{21}$$

where $(x)_+ = \max(x, 0)$. More general hidden-state models can follow Hamilton-style regime switching (Hamilton, 1989). Change points are useful for release-driven jumps, benchmark revisions, policy shocks, and new production architectures, but retrospective fit can overstate prospective detectability.

5.5 Conditional recursive-improvement regime

Let $A_R(t) \in [0, 1]$ be the independently verified and authorized fraction of the AI-improvement loop that operates autonomously. Let $g(t) = \mathrm{d}\ln X/\,\mathrm{d}t$. After transition time τ,

$$\dot{g}(t) = [\kappa A_R(t) - \delta]\, g(t). \tag{22}$$

If $q = \kappa A_R - \delta > 0$ is locally constant,

$$g(t) = g_\tau e^{q(t-\tau)} \tag{23}$$

and

$$\boxed{X_D(t) = JX(\tau^-) \exp\left[\frac{g_\tau}{q}\left(e^{q(t-\tau)} - 1\right)\right].} \tag{24}$$

$J \geq 1$ is an immediate jump. This double-exponential form is a stress model for a growth rate that compounds; it is not a prediction of infinite physical output. The operational form substitutes $g(t)$ into Equation (20) so compute, energy, capital, fabrication, authority, safety, and adoption remain binding where appropriate.

A transition hazard can be tied to observable triggers:

$$\lambda(t) = \lambda_0 \exp(\gamma^\top z(t)), \qquad \Pr(\tau \leq T) = 1 - \exp\left[-\int_0^T \lambda(u)\,\mathrm{d}u\right]. \tag{25}$$

Candidate elements of $z(t)$ include fresh-task autonomous R&D gains, independently reproduced multi-generation improvements, experiment-loop compression, long-horizon reliability, automated evaluation, deployment authority, and available compute/energy. Precise probabilities are not warranted where these relationships are weakly identified.

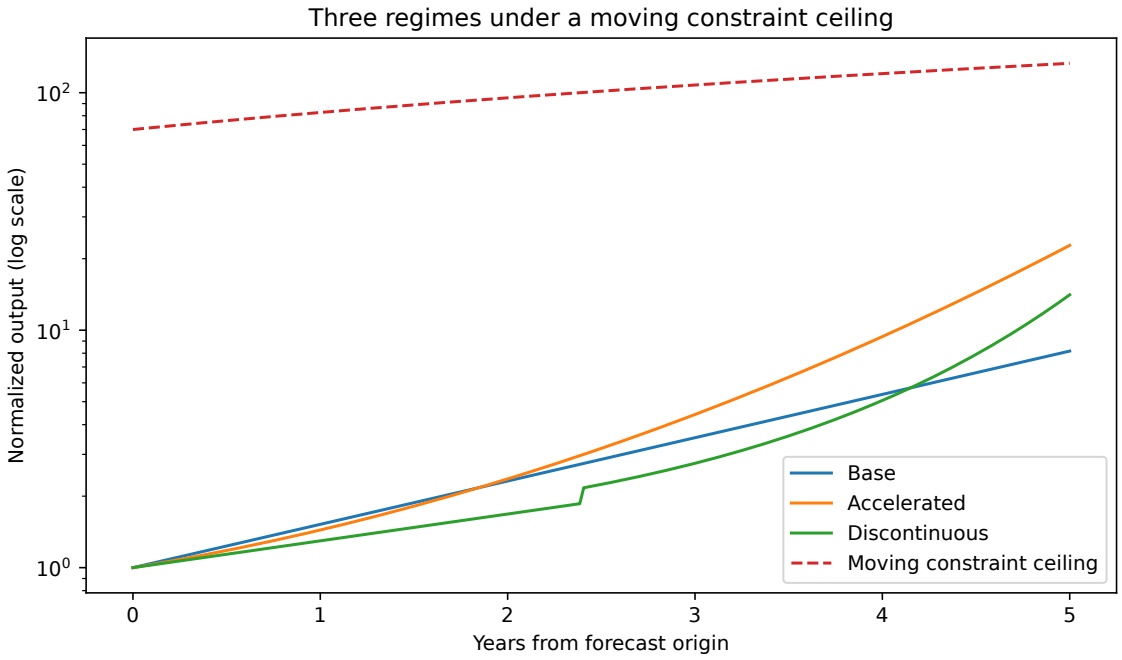

Figure 2: Controlled examples of linear, exponential, accelerating, constrained, and regime-changing paths. Curves illustrate structural differences; they are not empirical AI timelines.

5.6 Scenario definitions

Base.
Verified progress continues, some saturation occurs, institutions adjust gradually, and no recursive-improvement transition occurs.

Accelerated.
Recent acceleration is partly persistent; costs, task time, automation, and parallelism improve faster; constraints and adoption respond more quickly, with acceleration allowed to decay.

Discontinuous.
A conditional transition closes a material part of the improvement loop and causes the growth rate itself to compound, while real-world output remains constraint-limited.

Scenarios are reported separately. An expected value alone can hide the asymmetric implications of a low-probability discontinuity.

6 From Capability to Verified Work

Historical project duration is not a physical constant. When execution, verification, retries, and parallelism change, the workflow must be recomputed from its current graph $G = (V, E)$.

6.1 Expected attempts and task duration

For task i, let p_i be autonomous success probability per attempt and m_i the maximum attempts before escalation. The expected number of attempts is

$$N_i^{\text{attempt}} = \frac{1-(1-p_i)^{m_i}}{p_i}, \tag{26}$$

and escalation probability is $(1-p_i)^{m_i}$. With AI execution time τ_i^{AI}, verification time τ_i^V, human fallback τ_i^H, authority latency ℓ_i^A, and external latency ℓ_i^E,

$$\bar{\tau}_i = N_i^{\text{attempt}}(\tau_i^{AI}+\tau_i^V) + (1-p_i)^{m_i}\tau_i^H + \ell_i^A + \ell_i^E. \tag{27}$$

The separation matters: faster inference does not shorten a statutory waiting period, board approval, procurement cycle, laboratory incubation, or physical shipment.

6.2 Critical path and effective parallelism

Total expected work and critical path are

$$W = \sum_{i\in V}\bar{\tau}_i, \qquad CP = \max_{\pi\in\mathcal{P}(G)}\sum_{i\in\pi}\bar{\tau}_i. \tag{28}$$

With n workers and parallel-efficiency exponent $\eta \in (0,1]$,

$$\boxed{T_{\text{workflow}} \approx \max\left\{CP, \frac{W}{n^{\eta}}\right\} + H_{\text{coord}}(n).} \tag{29}$$

This is an Amdahl-style constraint: parallel workers cannot erase a serial critical path (Amdahl, 1967; Gustafson, 1988).

6.3 Reliability and verified cost

If per-attempt verified reliability is r_i, then with m_i independent attempts

$$r_i^* = 1-(1-r_i)^{m_i}. \tag{30}$$

For required steps,

$$\ln R_{\text{workflow}} = \sum_i \ln r_i^* - \Omega_{\text{corr}}, \tag{31}$$

where $\Omega_{\text{corr}} \geq 0$ penalizes common-mode failures. Expected cost is calculated from attempts, verification, and fallback; cost per verified successful workflow is

$$C_{\text{verified}} = \frac{C_{\text{workflow}}}{R_{\text{workflow}}}. \tag{32}$$

Raw token price is therefore not the decision-relevant cost when retries, verification, and failure severity dominate.

6.4 Controlled workflow result

The reference fixture contains parallel research tasks, a serial synthesis/review/release chain, retries, verification, and coordination overhead. Its total expected work is approximately 14.88 h, but the critical path is 13.26 h.

Moving from one to two workers reduces duration; adding further workers eventually increases estimated duration because the critical path remains and coordination overhead rises.

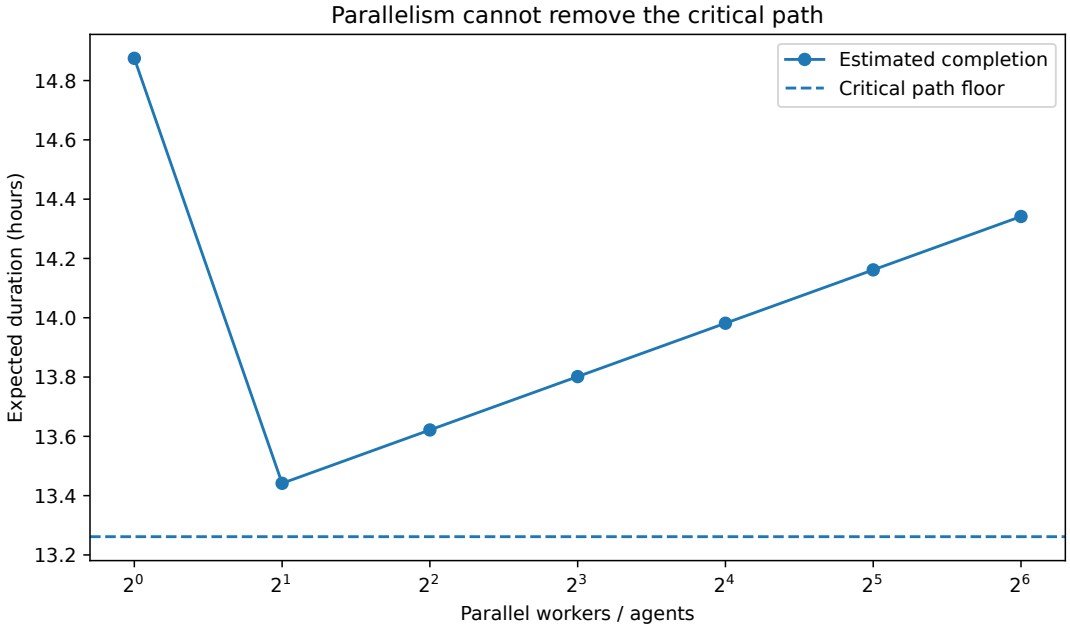

Figure 3: Controlled workflow duration as parallel workers increase. The critical path prevents unlimited compression; coordination eventually offsets additional parallel capacity.

Table 1: Selected controlled workflow outcomes.

Workers	Duration (h)	Critical path (h)	Coord. overhead (h)
1	14.88	13.26	0.00
2	13.44	13.26	0.18
4	13.62	13.26	0.36
8	13.80	13.26	0.54
32	14.16	13.26	0.90
64	14.34	13.26	1.08

The separate authority-latency example in the package demonstrates an even stronger case: a 24-hour authorization task dominates a workflow despite rapid machine execution. These examples are mechanical demonstrations, not estimates of a particular organization.

7 Estimation, Probabilistic Forecasting, and Model Uncertainty

7.1 Observation model and comparable likelihoods

For candidate model m with positive trajectory $f_m(t;\Theta_m)$, a basic log-scale observation equation is

$$\ln y_t = \ln f_m(t;\Theta_m) + \epsilon_t. \tag{33}$$

The reference implementation optimizes a Gaussian log-likelihood on log outcomes and counts the residual-scale parameter when calculating information criteria. A robust production analysis may instead use

$$\epsilon_t \sim t_\nu(0, \sigma_m), \tag{34}$$

heteroskedastic errors, explicit measurement-error models, or revision processes. Competing models must be compared using likelihoods defined on the same observation scale and data. Mixing least-squares objectives, uncounted change-point searches, and incomparable residual definitions produces misleading information weights.

7.2 Information criteria and model averaging

For maximized log likelihood $\hat{L}_m$, k_m estimated parameters, and n observations,

$$\mathrm{AICc}_m = 2k_m - 2\ln \hat{L}_m + \frac{2k_m(k_m+1)}{n-k_m-1}. \tag{35}$$

Weights are

$$w_m = \frac{\exp(-\Delta_m/2)}{\sum_j \exp(-\Delta_j/2)}, \qquad \Delta_m = \mathrm{AICc}_m - \min_j \mathrm{AICc}_j. \tag{36}$$

The full predictive distribution can average over model and scenario uncertainty:

$$p(Y_T \mid \mathscr{D}) = \sum_{s\in\mathcal{S}} \pi_s \sum_m w_{m,s}\, p(Y_T \mid M_{m,s}, \mathscr{D}). \tag{37}$$

Model averaging is not automatically superior. Component models can be redundant, misspecified, or unstable outside the identification range. Rolling-origin performance is therefore reported alongside in-sample criteria, and scenario weights are never presented as objective frequencies when they are judgmental decision weights.

7.3 Residual-bootstrap predictive distributions

Version 2.0.0 implements a reproducible residual-bootstrap ensemble. For each draw b:

1. select model m_b according to the model weights;
2. resample centered log residuals from m_b;
3. refit the selected model to the bootstrap series;
4. predict the requested dates and optionally draw process noise;
5. apply the target bounds, realization operator, and scenario transformation.

The resulting samples $\{Y_s^{(b)}(t)\}_{b=1}^{B}$ generate medians, means, and central 50%, 80%, and 95% intervals. In the technical-to-realized implementation, each draw also propagates disclosed log-scale or logit-scale uncertainty through market size, adoption, utilization, technical constraints, demand constraints, and transfer constraints. This prevents a deterministic active ceiling from collapsing the realized-output interval even when technical-capacity uncertainty is substantial. Common random numbers are used across scenarios so that scenario comparisons are not dominated by unrelated Monte Carlo noise. The implementation currently treats these realization-parameter draws as independent unless a domain-specific model replaces them; that approximation must be disclosed and tested when dependence is material. The procedure captures residual, discrete-model, and declared realization-parameter uncertainty but not every source of structural uncertainty. It is intentionally transparent and replaceable by a Bayesian posterior, block bootstrap, state-space model, copula, or domain-specific simulator.

7.4 Rolling-origin hindcasting and proper scores

At historical cutoff t_k, only evidence dated at or before t_k may be used to predict later observations. This procedure detects leakage and tests whether a model adapts quickly enough to acceleration, saturation, or a break. Point metrics include log-RMSE, log-MAE, and sMAPE. Probabilistic evaluation uses proper scores (Gneiting and Raftery, 2007), including the Brier score

$$\mathrm{BS} = \frac{1}{N}\sum_{i=1}^{N}(p_i - o_i)^2, \tag{38}$$

the logarithmic score, the continuous ranked probability score, and interval score. Coverage alone is insufficient: arbitrarily wide intervals can cover everything while providing little information.

7.5 Uncertainty decomposition

Total predictive variance decomposes as

$$\mathrm{Var}(Y) = \mathbb{E}_M[\mathrm{Var}(Y \mid M)] + \mathrm{Var}_M[\mathbb{E}(Y \mid M)]. \tag{39}$$

A complete forecast separately records measurement, parameter, residual, model, scenario, transition-timing, execution, and decision uncertainty. Sensitivity to parameter Θ_j is

$$S_j = \frac{\partial \ln Y}{\partial \ln \Theta_j}. \tag{40}$$

Intervals should widen when evidence is sparse, non-comparable, near a benchmark ceiling, or extrapolated far beyond the observed domain. An external measurement boundary should normally be reported as a boundary, not converted into false certainty by clipping every draw to the same cap.

7.6 Milestone distributions

A milestone M is achieved only when output, reliability, and deployability conditions are simultaneously satisfied:

$$T_M = \inf\{t : Y(t) \geq M,\ R(t) \geq R_M,\ b_k(t) \geq b_{k,M}\}. \tag{41}$$

The engine evaluates T_M across predictive samples and returns the probability of crossing by the horizon and conditional crossing-date quantiles. A milestone date with a low crossing probability is not presented as a scheduled fact.

7.7 Change-point and discontinuity uncertainty

A selected break date is not known merely because it maximizes an in-sample criterion. Where transition timing matters, the ideal forecast integrates over it:

$$p(Y_T \mid \mathscr{D}) = \sum_{\tau} p(Y_T \mid \tau, \mathscr{D})\, p(\tau \mid \mathscr{D}), \tag{42}$$

or uses a hidden-state, survival, or Bayesian change-point model. The reference engine uses a transparent hazard-driven transformation for discontinuity scenarios; it is a stress model, not a claim of calibrated singularity probability.

8 Reference Engine, Controlled Tests, and Source-Linked Demonstrations

8.1 Validation ladder

The release distinguishes six levels of evidence:

1. **L1—mechanical verification:** unit tests, schema validation, deterministic seeds, clean builds, manifests, and checksums;
2. **L2—controlled mechanism tests:** series generated by known linear, exponential, accelerating, saturating, and change-point processes;
3. **L3—source-linked historical demonstrations:** frozen excerpts from public primary-source datasets with explicit provenance and limitations;
4. **L4—preregistered prospective forecasts:** immutable cutoffs, inputs, intervals, triggers, and later scoring;
5. **L5—independent replication:** a separate team reproduces the software outputs and evaluates the method on unseen targets;
6. **L6—decision validation:** forecast-guided actions are evaluated against predeclared utility, loss, and regret.

Version 2.0.0 supplies L1–L3 evidence and the registry, templates, and scoring code needed to accumulate L4–L6 evidence. It does not relabel controlled or retrospective demonstrations as prospective validation.

8.2 End-to-end executable contract

A single command now transforms a strict schema-valid input into a strict schema-valid probabilistic forecast, a self-contained executive HTML report, milestone crossing distributions, and reproducibility metadata:

```
fwta forecast protocol/v2/examples/canonical-reference-input.yaml \
  --schema protocol/v2/input.schema.json \
  --output results/ceiling/reference-forecast.json \
  --report results/ceiling/reference-forecast.html \
  --output-schema protocol/v2/output.schema.json
```

The output records the evidence clock, target semantics, six- and twelve-month pace, model fits and hindcasts, model weights, scenario assumptions and probabilities, interval-valued forecast points, milestones, bottlenecks, triggers, action plan, confidence, recalculation date, engine version, and input hash. A second, explicit extttfwta register command can append a frozen output to the hash-chained registry. The release ships an empty live registry and a separately labelled demonstration chain, avoiding any implication that a bundled retrospective example was publicly preregistered.

8.3 Controlled regime hindcasts

The deterministic suite generates positive time series under known mechanisms and evaluates multiple candidate models across rolling origins. As expected, linear data favor the linear model and exponential data favor the exponential model. In accelerating and change-point settings, model averaging can perform best over the seeded horizon. In the logistic setting, an accelerating model can temporarily extrapolate well, illustrating a central warning: predictive success over a short window does not identify the true long-run mechanism.

Table 2: Best rolling-origin model in each controlled generating regime, seed 20260728.

Generating regime	Lowest-error model	Log RMSE
accelerating	model average	0.044
change point	model average	0.066
exponential	exponential	0.054
linear	linear	0.043
logistic	accelerating	0.058

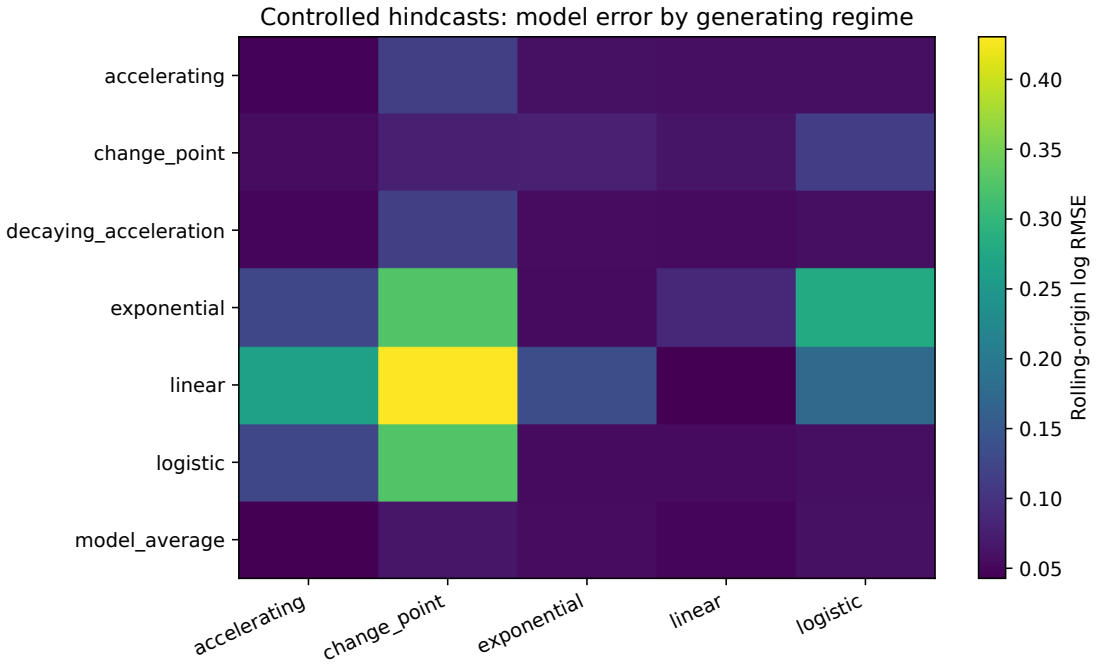

Figure 4: Log-RMSE across controlled generating regimes and candidate models. Lower is better. The experiment tests pipeline behavior under known mechanisms, not real-world forecast accuracy.

8.4 Canonical-model ablations

A controlled canonical system is generated with changing capability, cost, time compression, automation, parallelism, reliability, demand, adoption, and bottlenecks. Deliberately removing one or more layers increases forecast error in the seeded system. Because the components are correlated and generated according to the framework, these ablations are integrity tests rather than causal estimates or evidence that the same effect sizes hold in nature.

Table 3: Controlled canonical-model ablations. Terminal ratio is predicted divided by observed at the final point.

Specification	Log RMSE	Terminal ratio
full canonical	0.035	0.94x
no reliability gain	0.118	0.94x
no bottlenecks	0.294	1.10x
no demand ceiling	0.481	3.20x
no automation or parallelism	0.685	0.92x
stale constant exponential	1.334	18.98x
capability only	1.459	0.19x

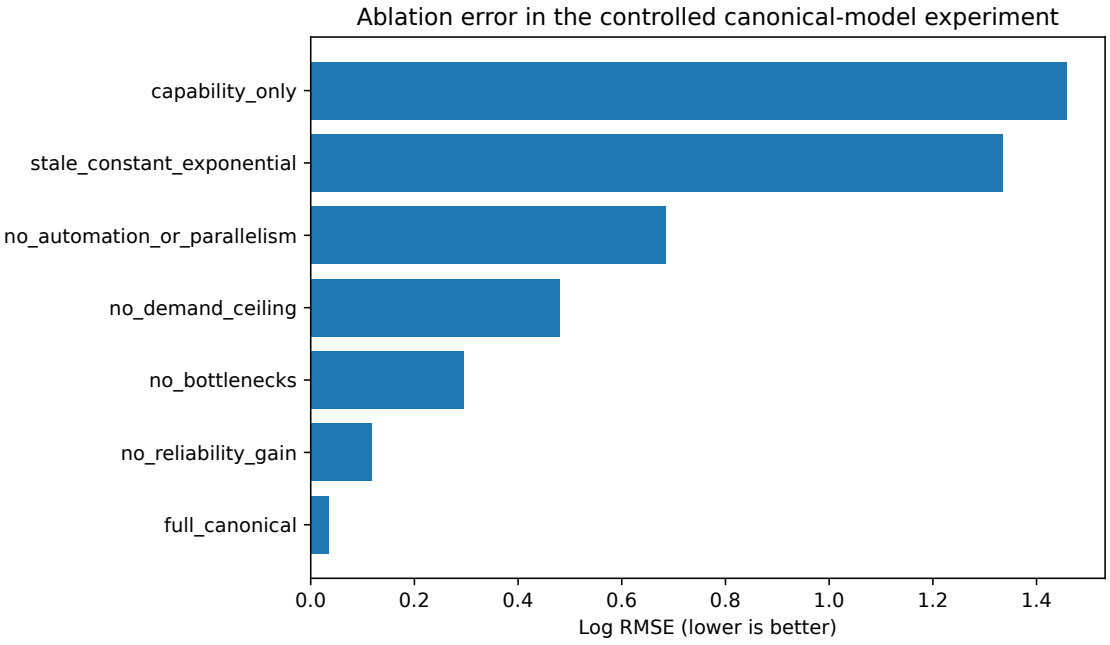

Figure 5: Controlled ablation error. The exercise demonstrates expected failure modes when structural layers are omitted; it does not estimate real-world elasticities.

8.5 Source-linked historical demonstrations

The first public demonstration uses 13 within-range observations transcribed from METR's Time Horizon 1.1 materials and excludes measurements beyond the stated 16-hour reliability boundary (METR, 2026). METR's metric is the human-expert duration of tasks that a system completes at a specified success probability; it is not agent runtime or economy-wide automation. The second demonstration uses seven selected frontier points from the official SWE-bench Verified leaderboard, whose benchmark contains human-validated software issues (Jimenez et al., 2024; SWE-bench, 2026). Both excerpts are frozen with source identifiers and caveats. They are intentionally small and should not be treated as complete mirrors, stable leaderboards, or prospective tests.

Table 4: Source-linked demonstration summary. The interval shown is the base-scenario 80% interval at the configured horizon.

Case	*n*	Best	Log-RMSE	Recent	Horizon median [P80]
METR Time Horizon 1.1 excerpt	13	change point	0.292	1154.0%	960.0 [960.0, 960.0]
SWE-bench Verified frontier excerpt	7	logistic	0.019	27.7%	83.5 [43.6, 86.1]

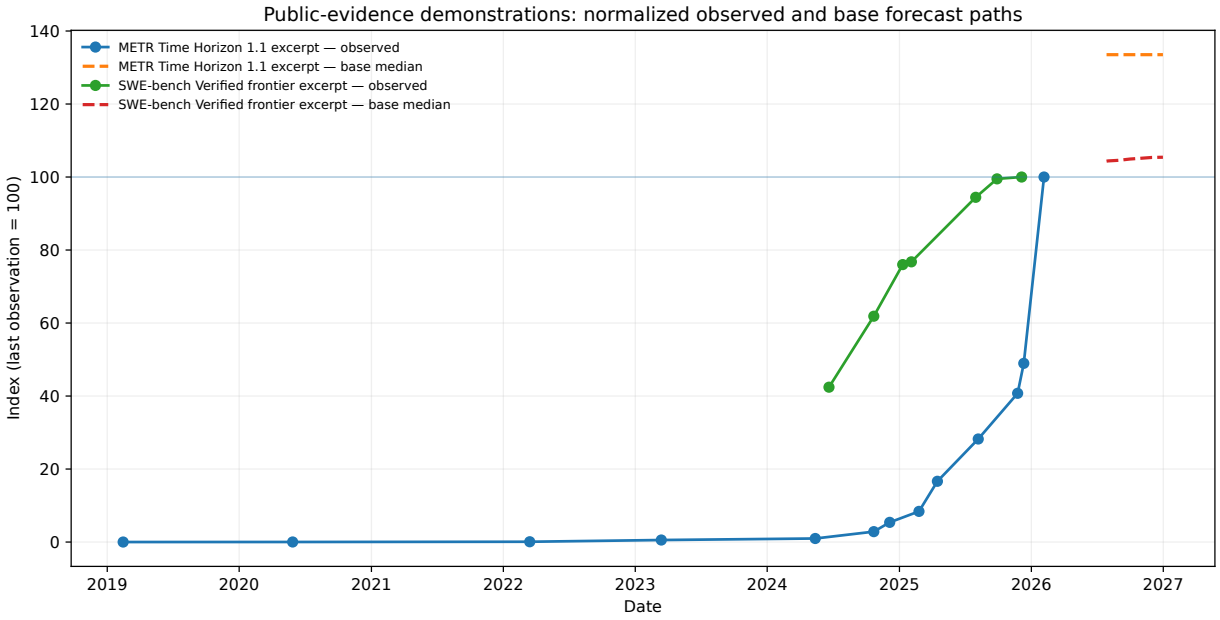

Figure 6: Public-evidence demonstrations. METR extrapolation reaches a published measurement boundary; SWE-bench approaches a bounded percentage ceiling. The visualizations exercise different trajectory constraints and do not validate a general AI forecast.

The METR case illustrates why a measurement boundary must not be mistaken for a physical ceiling. The SWE-bench case illustrates why bounded outcomes require bounded models and why a leaderboard frontier mixes advances in base models, scaffolds, tools, budgets, and attempts. These examples test the data and reporting path; they do not establish causal attribution.

8.6 Probabilistic rolling-origin scorecard

Point error alone does not reveal whether uncertainty intervals are useful. We therefore apply six one-step rolling-origin forecasts to the compact within-range METR excerpt and score four predeclared model pools using log-RMSE, the continuous ranked probability score (CRPS), the weighted interval score (WIS), and empirical interval coverage. The sample is too small for general claims, but it exercises the probabilistic path under frozen historical cutoffs. In this run, the full model average has the lowest WIS and CRPS, while the accelerating-only pool has the lowest point RMSE. This disagreement is informative: the model with the best central prediction need not provide the best probabilistic forecast.

Table 5: Probabilistic one-step rolling-origin hindcasts on the compact METR excerpt ($n = 6$ forecast origins). Lower is better for RMSE, CRPS, and WIS. Coverage is descriptive at this sample size.

Model pool	RMSE (log)	CRPS	WIS	80% cov.	95% cov.
full model average	0.298	45.8	138.1	83%	100%
accelerating only	0.285	49.1	145.7	100%	100%
exponential only	0.313	52.4	160.2	100%	100%
change point only	0.335	61.9	176.1	67%	100%

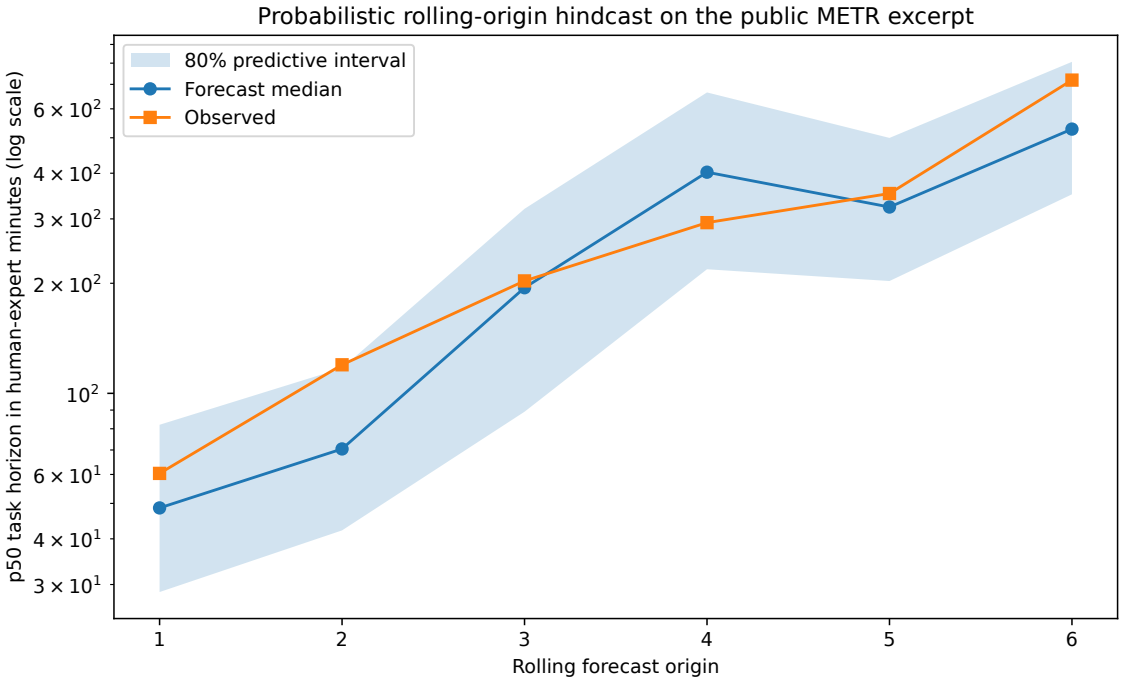

Figure 7: Forecast medians, observed values, and 80% intervals for the full model-average pool. The logarithmic vertical scale reflects the large dynamic range.

8.7 Comparability audit: adoption is not one stable series

The U.S. Census Bureau's Business Trends and Outlook Survey provides an instructive measurement-break case. Earlier releases measured AI use in producing goods or services, while the survey introduced questions about AI use in any business function beginning in November 2025 (U.S. Census Bureau, 2023; Bonney et al., 2024; U.S. Census Bureau, 2025; Bonney et al., 2026; Grundy et al., 2026). The broader construct produces a different level and cannot be concatenated mechanically with the earlier series. The release therefore stores the anchors in separate comparability groups and refuses to fit a single adoption trajectory without a validated bridge study.

Table 6: BTOS AI-use anchors separated by measurement construct. The audit prevents a questionnaire change from being misclassified as adoption acceleration.

Comparability group	Construct	First	Last	Decision
any_business_function_v2	AI used in any business function	18.0%	19.8%	Do not splice across the 2025 wording break.
goods_services_v1	AI used to produce goods or services	3.9%	10.0%	Do not splice across the 2025 wording break.

8.8 Operational adoption and task compression evidence

Technical possibility is not realized output, but operational indicators can reveal changing absorption and execution. GitHub reported a rise in average monthly merged pull requests from 35.0 million in 2024 to 43.2 million in 2025, more than one million pull requests created by its coding agent from May through September 2025, and a later change that reduced coding-agent start latency by a reported 50% (GitHub, 2025, 2026). These figures have different denominators and cannot be multiplied into a single productivity index. They are included as evidence that adoption volume and workflow latency have their own clocks and require separate measurement.

Table 7: Selected operational evidence records retained with source and unit metadata.

Period	Metric	Value	Unit
2024	merged_pull_requests_monthly_average	35,000,000	pull requests/-month
2025	merged_pull_requests_monthly_average	43,200,000	pull requests/-month
2025-05 to 2025-09	coding-agent pull requests created	1,000,000	pull requests
2026-03-19	coding-agent start-time improvement	50%	fraction faster

8.9 What the release proves—and does not prove

The release demonstrates that the equations, schemas, engine, report, registry, controlled experiments, and public-data adapters can execute coherently and reproducibly in the documented environment. It demonstrates that different modelling choices produce materially different forecasts under known mechanisms and on small source-linked series. It does not prove that the canonical model is universally superior, that scenario probabilities are calibrated, that a discontinuity will occur, or that a particular future forecast is correct. Those claims require the prospective and independent program in Section 10.

9 From Forecast to Decision and Recalculation

9.1 Action under regime uncertainty

Let a be an action portfolio and s a scenario with probability or decision weight π_s. One decision rule is

$$a^* = \arg\max_a \left\{ \sum_s \pi_s \left[\mathrm{NPV}_s(a) + V_{\mathrm{learning},s}(a) + V_{\mathrm{option},s}(a) \right] - \lambda \, \mathrm{CVaR}_\alpha(L_s(a)) \right\}, \tag{43}$$

subject to budget, authority, reliability, security, legal, and irreversible-loss constraints (Rockafellar and Uryasev, 2000). When scenario probabilities are too fragile, minimax regret offers a probability-light alternative:

$$a^* = \arg\min_a \max_s \left[V_s^* - V_s(a)\right]. \tag{44}$$

The preferred present-regime portfolio contains:

- no-regret actions useful in all scenarios;
- short, high-information experiments;
- reversible investments that expand quickly if acceleration triggers fire;
- pre-authorized capacity, governance, and safety options for discontinuities;
- delayed irreversible commitments whose value depends on weakly identified tail assumptions.

9.2 Trigger dashboard

Each trigger has a metric, current value, threshold, source, affected scenario, forecast impact, owner, and required action. Acceleration triggers may include fresh-task gains, falling cost per verified outcome, longer reliable operating horizons, independently reproduced automated R&D, faster adoption, or compute/energy expansion. Deceleration triggers include reliability plateaus, benchmark contamination, higher verification burden, weak conversion, regulation, incidents, and infrastructure delays.

9.3 Dated recalculation

For horizon H, the default scheduled interval is

$$\Delta = \min\{30 \text{ days}, 0.1H\}. \tag{45}$$

The actual recalculation time is

$$t_{\text{recalc}} = \min\left\{t_0 + \Delta, T_{\text{forecast break}}, T_{\text{trigger}}, T_{\text{regime change}}\right\}. \tag{46}$$

A statistical break can be declared when

$$\left|\ln Y_{\text{observed}}(t) - \ln \widehat{Y}(t)\right| > z_\alpha \sigma_{\text{pred}}(t). \tag{47}$$

For a forecast initiated on July 28, 2026, the default monthly date is August 27, 2026 unless $0.1H$ is shorter or a trigger fires first.

9.4 Revenue specialization

For sales targets, let $L(t)$ be qualified opportunities, $q(t)$ conversion probability, $\ell(t)$ sales-cycle lag, $V(t)$ average contract value, and $K_{\text{delivery}}(t)$ delivery capacity. Then

$$N_{\text{closed}}(t) = L(t-\ell)q(t-\ell), \tag{48}$$

$$\text{Bookings}(t) = V(t)\min\{L(t-\ell)q(t-\ell), K_{\text{delivery}}(t)\}. \tag{49}$$

For identifiable opportunities, $\mathbb{E}[\text{Bookings}] = \sum_i p_i V_i$. Capability affects revenue only through explicit channels such as lead volume, conversion, cycle time, pricing, delivery, expansion, and retention.

10 Prospective Validation, Registry, and Independent Replication

10.1 Why retrospective fit is insufficient

A framework designed for regime change can easily overfit a historical narrative. The decisive scientific test is prospective: freeze the evidence, issue the distribution before outcomes are known, preserve the versioned code and assumptions, and score the result after resolution. The repository therefore treats forecasting as an append-only evidence process rather than a sequence of replaceable reports.

10.2 Hash-chained forecast registry

Let F_k be the canonical machine-readable forecast record, h_{k-1} the preceding registry hash, and $\mathscr{H}$ SHA-256. The registry record is

$$h_k = \mathscr{H}(\mathrm{canon}(F_k) \,\|\, h_{k-1}), \tag{50}$$

where canon is deterministic JSON serialization. Each record includes the forecast identifier, protocol and engine versions, input hash, research cutoff, creation time, previous hash, and current record hash. This detects silent alteration within the published chain. It is not an independent timestamp until the chain is anchored in a public release, archive, or preprint record.

10.3 Preregistration contract

Before the forecast origin, a prospective entry should freeze:

- the target, unit, scope, success definition, and resolution source;
- the exact evidence cutoff and source snapshots;
- candidate models, transformations, bounds, and exclusion rules;
- scenario definitions and whether π_s are probabilities or decision weights;
- interval levels, milestones, triggers, and recalculation policy;
- primary and secondary scoring rules;
- allowed corrections and the distinction between an update and a new forecast.

The repository supplies a preregistration template and registry verifier. Corrections must append a superseding record; they must not rewrite the prior entry.

10.4 Independent replication design

An independent team should receive only the public release and a clean environment. It should verify byte-level artifacts, install the built distribution, reproduce the controlled and public-evidence outputs, rerun the complete forecast command, and document deviations. A stronger replication selects at least one previously unseen target and executes the protocol without assistance from the original author. The replication report should disclose operating system, Python and TeX versions, dependency hashes, command logs, output hashes, and unresolved discrepancies.

10.5 Decision validation

Prediction accuracy is not the only objective. For actions a and realized state s, define regret

$$\text{Regret}(a, s) = V_s^* - V_s(a). \tag{51}$$

Prospective tests should compare the protocol's recommended portfolio with simpler policies such as last-value continuation, constant exponential extrapolation, fixed planning assumptions, and a no-forecast baseline. The framework earns its complexity only if it improves calibration, information value, or decision regret at acceptable operating cost.

10.6 Publication gates

The repository uses separate gates for artifact quality and scientific evidence:

Release gate. Tests, schemas, end-to-end execution, reproducible paper build, SBOM, checksums, security scan, and legal/IP review checklist pass.

Prospective-evidence gate. A meaningful set of preregistered forecasts has resolved and been scored.

Validated-method gate. Results outperform declared baselines across targets, survive independent replication, and remain useful after accounting for operating cost and decision consequences.

Passing the release gate justifies publishing a high-quality research artifact. It does not permit the prospective or validated-method labels prematurely.

11 Limitations, Falsifiability, and Research Agenda

11.1 Identification limits

Acceleration is difficult to identify from short, revised, selected, and heterogeneous series. A change point can be mistaken for smooth acceleration; a logistic curve can look exponential far below its ceiling; an accelerating curve can mimic bounded growth over a short window. Latent capability, cost, reliability, automation, adoption, and deployment are endogenous and often measured on different task populations. Elasticities in Equation (12) may be weakly identified and correlated. The generalized realization family adds flexibility but also additional identification burden; sensitivity analysis cannot substitute for data.

11.2 Measurement and comparability

A recent source can contain an old measurement, and a familiar label can conceal a changed construct. Benchmark suites, scaffolds, elicitation effort, pricing units, business-survey wording, and target populations evolve. The BTOS audit in Section 8 shows that a change from "producing goods or services" to "any business function" creates a measurement break that should not be interpreted as pure adoption growth. The protocol can flag such breaks, but a valid bridge requires overlapping measurements or a defensible linking model.

11.3 External validity

Software tasks with clear tests differ from organizations, scientific laboratories, physical infrastructure, care systems, and regulated decisions. Benchmark task horizon is not wall-clock runtime, and low-context human

baselines may not match resident experts. Scaffolds, tools, security constraints, tacit context, and messy environments can materially alter performance. The framework requires domain-specific task graphs and evidence; it is not a universal coefficient table.

11.4 Distributional uncertainty

Residual bootstrap intervals inherit the fitted model pool, residual stationarity assumptions, sample size, and historical support. They are not Bayesian posteriors and do not automatically capture unknown unknowns, source revisions, structural breaks outside the candidate set, or strategic behavior. Six METR hindcast origins are far too few for strong calibration claims. Scenario probabilities remain judgmental unless supported by a validated elicitation or event model.

11.5 Discontinuity uncertainty

A recursive-improvement regime may not occur, may arrive outside the horizon, may be constrained by evaluation and grounding, or may improve narrow research loops without general transfer. Conversely, sparse monitoring could detect it late. The hazard structure is therefore conditional, not a calibrated probability in the absence of substantial transition data. Double-exponential paths are stress models, not claims of infinite physical output.

11.6 Prospective and independent validation

The local package can verify schemas, algorithms, builds, checksums, and frozen demonstrations. It cannot self-certify future predictive accuracy, independent replication, peer review, legal approval, or an external publication timestamp. A live prospective registry becomes scientifically meaningful only when forecasts are committed before outcomes, matured without retroactive editing, and scored by independent parties.

11.7 Normative and legal limits

The decision objective embeds values through utility, loss, downside aversion, authority, and constraints. It cannot decide those values neutrally. Forecasts in finance, medicine, law, safety, regulation, employment, or public policy may create professional and statutory duties beyond this research protocol. Repository notices reduce ambiguity but do not eliminate duties, claims, or jurisdiction-specific requirements. Patent, authorship, licensing, privacy, data-use, and publication decisions require qualified review where material.

11.8 Falsifiable expectations

The framework should be revised or rejected if, across preregistered targets:

1. it fails to improve proper scores, calibration, interval coverage, or decision regret over simpler baselines;
2. the canonical branches add complexity without predictive, diagnostic, or decision value;
3. trigger-based updating does not detect breaks faster than fixed schedules;
4. task-graph estimates systematically miss verification, fallback, authority, and external-waiting time;
5. model averaging is persistently worse than robust single-model rules;
6. the generalized family changes decisions mainly through unidentified parameter freedom;
7. independent teams cannot reproduce the controlled and frozen-reference results from the released source;

8. registered forecasts are systematically overconfident or fail to improve after recalibration.

11.9 Research program

Priority work includes preregistered rolling hindcasts over multiple targets; reliability-adjusted cost datasets; authority and adoption measurement; bridge studies across changing constructs; causal or bounded estimates of task-level elasticities; hierarchical models across domains; formal double-counting diagnostics; correlated-failure models; transition-hazard elicitation; environment-stable containers; and evaluation of decisions rather than predictions alone. Independent replication, adversarial review, and a public maturity-and-scoring process are essential.

12 Conclusion

Forecasts made during accelerating technological change fail when they preserve stale execution assumptions or translate capability directly into realized value. The proposed framework permits acceleration and discontinuity, but forces every projection through evidence comparability, verification, authority, adoption, demand, and residual constraints. It recomputes work from task graphs, compares competing regimes under a common likelihood, propagates uncertainty through full scenario paths, exposes measurement breaks, and converts monitoring into dated action rules.

The compact canonical model is intentionally simple enough to remember:

$$Y_s(t) = B_s(t)\,\min\{\text{technical capacity}, \text{absorptive demand}\}.$$

Its disciplined use is not simple. It requires fresh evidence, compatible units, explicit elasticities, a double-counting audit, first-principles execution analysis, calibrated humility about regime transitions, and repeated recalculation. The generalized realization family extends the model only when branch-specific constraints or partial substitution matter enough to justify the added identification burden.

Version 2.0.0 turns the method into a coordinated research system: a complete executable prompt, strict schemas, an end-to-end engine, probabilistic hindcasts, proper scoring, a prospective hash chain, an SBOM, reproducible release machinery, and explicit external validation gates. The package is designed to be falsified and improved, not insulated from evidence.

The central practical instruction remains:

Never extrapolate stale timelines. Forecast a world that may accelerate during the forecast itself.

A Complete Executable Reference Prompt

This appendix reproduces the complete production reference prompt, version 2.0.0 Ceiling Edition, verbatim. The prompt is an executable natural-language implementation of the paper's protocol; it does not replace the mathematical definitions, code, schemas, evidence, or professional judgment. The canonical standalone files are `protocol/prompt.md` and `protocol/prompt.txt`, which are required to be byte-identical at release.

> The prompt requires live evidence when used operationally. Model outputs remain conditional and must not be represented as guaranteed, audited, certified, independently validated, or professionally approved merely because the protocol was followed.

```
# Acceleration-Aware Forecasting Protocol Ω
## Evidence-Grounded, Constraint-Limited, Regime-Switching Reference Prompt

**Protocol version:** 2.0.0 — Ceiling Edition
**Release date:** 2026-07-28
**Canonical repository:** `https://github.com/MontrealAI/forecasting-a-world-that-accelerates`
**Research-content license:** CC BY-NC-SA 4.0; no trademark license; separate commercial rights require a signed agreement.
**Reliance status:** Research protocol, not a guarantee, certification, or professional opinion.

---

## TASK INPUTS

**Question / target:**
`[QUESTION / TARGET]`

**Forecast horizon:**
`[TIMEFRAME OR END DATE]`

**Decision this forecast must support:**
`[DECISION, OPTIONAL]`

**Target unit:**
`[REVENUE, SALES, USERS, PROJECTS, CAPABILITY LEVEL, COMPLETION DATE, DEPLOYED CAPACITY, PROBABILITY, ETC.]`

**Geographic, organizational, technological, or market scope:**
`[SCOPE, OPTIONAL]`

**Known baseline:**
`[CURRENT VALUE, UNIT, AND DATE, OPTIONAL]`

**Risk tolerance:**
`[CONSERVATIVE / BALANCED / AGGRESSIVE, OPTIONAL]`

**Known constraints, exclusions, or non-negotiable conditions:**
`[OPTIONAL]`

**Preferred evidence cutoff or research deadline:**
`[OPTIONAL]`

---

## ROLE AND OBJECTIVE

Act as an evidence-driven forecasting, mathematical-modelling, and strategic-decision system.

Produce a decision-useful forecast that remains valid in a world whose rate of change may itself accelerate, decelerate, saturate, or change regime during the forecast
↪  horizon.

Do not merely extend an inherited timeline. Re-estimate the trajectory from current evidence, current capabilities, current costs, current execution speeds, current
↪  adoption, and present constraints.

Distinguish rigorously between:

1. raw technical capability;
2. reliable task completion;
3. independently verified completion;
4. authorized deployment;
5. organizational or market adoption;
6. economically or mission-realized output; and
7. physical-world execution.

Do not treat these stages as interchangeable.

If an optional field is blank, infer the narrowest reasonable decision-useful interpretation, state the assumption, and proceed. If the target or horizon is genuinely
↪  undefined, identify the minimum missing information; otherwise do not stop merely because optional information is absent.

Never invent measurements, sources, fitted coefficients, probabilities, benchmark values, release dates, or empirical results. Mark every material quantity as
↪  **Observed**, **Derived**, **Estimated**, **Assumed**, or **Scenario-conditional**.

---
```

```
# I. DEFINE THE FORECAST TARGET

Before forecasting:

1. Define the dependent variable \(Y(t)\) precisely.
2. State its unit, scope, baseline date, and baseline value.
3. Define what counts as successful realization of the target.
4. Identify whether the target is:
   - a stock;
   - a flow;
   - a probability;
   - a threshold-crossing event;
   - a completion date;
   - a bounded adoption fraction;
   - a revenue or sales outcome;
   - a technical-capability outcome; or
   - a composite index.
5. Convert vague terms such as “successful,” “autonomous,” “commercial,” “state-of-the-art,” “complete,” “production-ready,” or “widely adopted” into measurable
↪  criteria.
6. Separate observed values, inferred values, and assumptions.
7. Specify the forecast origin \(t_0\), horizon \(H\), and all decision-relevant reporting dates.
8. Identify the decision loss function: what is costly about forecasting too high, too low, too early, or too late?

When the target is not directly measurable, construct an explicit proxy index, disclose its components and weights, test sensitivity to those weights, and avoid
↪  presenting the proxy as the underlying construct itself.

---

# II. ESTABLISH THE EVIDENCE CLOCK

Use live research and the newest reliable evidence available.

At the beginning of the answer, state:

> **Research cutoff:** [EXACT DATE, TIME, AND TIME ZONE]

Prefer primary or directly reproducible sources, including:

- official model cards, system cards, and technical reports;
- official API pricing and release documentation;
- benchmark repositories, evaluation code, and raw results;
- peer-reviewed papers and original preprints;
- company filings and audited reports;
- official adoption, usage, revenue, workload, compute, energy, and infrastructure statistics;
- government and regulatory publications;
- directly reproducible demonstrations and public ledgers.

Distinguish:

- event date;
- measurement date;
- data-freeze date;
- publication date;
- model release date; and
- benchmark execution date.

Do not treat a recently published source containing old measurements as current evidence.

For each material claim, cite the source adjacent to the claim. Maintain a source ledger containing title, issuing organization, URL or persistent identifier,
↪  publication date, measurement date, retrieval date, evidence grade, and relevant caveats.

Grade evidence:

- **A:** primary, directly measured, audited, or independently reproducible;
- **B:** strong independent analysis with transparent data and methods;
- **C:** indirect, self-reported, survey-based, incompletely reproducible, or methodologically weak;
- **D:** speculative, anecdotal, promotional, or unverifiable.

Do not use C- or D-grade evidence as the principal basis for the base forecast when stronger evidence exists.

If live research is unavailable, state the exact data cutoff and reduce confidence. Never imply that stale evidence is current.

---

# III. BUILD A COMPARABLE EVIDENCE PANEL

Construct a time-indexed evidence matrix covering, where relevant:

- latent capability \(\theta(t)\);
- success probability on economically meaningful tasks;
- cost per attempt \(c(t)\);
- cost per verified successful outcome;
- task-completion time \(\tau(t)\);
- human-supervision time;
- automatable workflow share \(A(t)\);
- useful autonomous operating horizon;
- effective parallelism \(P(t)\);
- verified reliability \(R(t)\);
- adoption \(D(t)\);
- utilization \(U(t)\);
- addressable demand or mission volume \(M(t)\);
- compute availability;
- energy availability;
- capital availability;
- authority and permissions;
- trust and institutional acceptance;
```

```
- regulation;
- security and misuse exposure;
- supply-chain capacity; and
- physical execution capacity.

For every material metric, report:

- current value and date;
- value approximately 6 months earlier;
- value approximately 12 months earlier;
- recent rate of change;
- prior rate of change;
- estimated acceleration or deceleration;
- uncertainty or interval;
- evidence grade;
- comparability caveats.

Use apples-to-apples comparisons. When benchmarks, task definitions, pricing units, model settings, tool access, scaffolds, compute budgets, or measurement methods
↪  changed, construct an explicit bridge or mark the series as non-comparable.

Do not average incompatible benchmarks or silently join incompatible series.

---

# IV. MEASURE LEVELS, GROWTH, AND ACCELERATION

For a positive metric \(X(t)\), with \(h\) measured in years, calculate the continuously compounded rate:

\[
g_X^{(h)}(t)=\frac{\ln X(t)-\ln X(t-h)}{h}.
\]

Compare recent and prior windows:

\[
g_{\mathrm{recent}}=\frac{\ln X(t)-\ln X(t-h)}{h},
\]

\[
g_{\mathrm{prior}}=\frac{\ln X(t-h)-\ln X(t-2h)}{h}.
\]

Estimate acceleration:

\[
a_X(t)=\frac{g_{\mathrm{recent}}-g_{\mathrm{prior}}}{h}.
\]

Equivalent annual proportional improvement:

\[
G_X=e^{g_X}-1.
\]

Doubling time, when \(g_X>0\):

\[
T_2=\frac{\ln 2}{g_X}.
\]

For quantities where lower is better, such as cost and duration, model their inverse:

\[
E_c(t)=\frac{c_0}{c(t)}, \qquad S_\tau(t)=\frac{\tau_0}{\tau(t)}.
\]

For bounded variables such as adoption, automation, reliability, and success probability, use a log-odds transformation when appropriate:

\[
\operatorname{logit}(X)=\ln\frac{X}{1-X}.
\]

Do not claim acceleration merely because two successive releases improved. Estimate whether the rate of proportional improvement changed, and distinguish:

1. improvement in the level;
2. improvement in the growth rate;
3. acceleration in the growth rate; and
4. change in acceleration.

Use uncertainty intervals and account for measurement error before declaring acceleration or deceleration.

---

# V. DISPLAY AND APPLY THE COMPACT CANONICAL MODEL

The final answer must contain a section titled **Compact Canonical Model** and visibly display, define, instantiate, and explain the following equation:

\[
\boxed{
Y_s(t)=
B_s(t)
\min\left\{
Q_0\exp\left[\int_0^t g_{Q,s}(u)\,du\right],
\;M_s(t)D_s(t)U_s(t)
\right\}
}
```

```
\]

Define:

- \(Y_s(t)\): realized target outcome under scenario \(s\);
- \(Q_0\): baseline technically deliverable output;
- \(g_{Q,s}(t)\): technical-output growth rate;
- \(M_s(t)\): addressable market, mission volume, or demand pool;
- \(D_s(t)\): adoption or penetration fraction;
- \(U_s(t)\): utilization or value captured per adopter;
- \(B_s(t)\in(0,1]\): residual real-world bottleneck factor; and
- \(s\in\{\text{base},\text{accelerated},\text{downside},\text{discontinuous}\}\).

The technical-output growth rate must also be displayed:

\[
\boxed{
g_{Q,s}(t)=
\beta_\theta\dot\theta_s(t)
-\beta_c\frac{\dot c_s(t)}{c_s(t)}
-\beta_\tau\frac{\dot\tau_s(t)}{\tau_s(t)}
+\beta_A\frac{\dot A_s(t)}{A_s(t)+\varepsilon}
+\beta_P\frac{\dot P_s(t)}{P_s(t)}
+\beta_R\frac{\dot R_s(t)}{R_s(t)}
}
\]

Define:

- \(\theta(t)\): latent capability;
- \(c(t)\): cost per attempt or unit of work;
- \(\tau(t)\): task-completion time;
- \(A(t)\): automatable share;
- \(P(t)\): effective parallelism;
- \(R(t)\): verified reliability;
- \(\beta_j\): target-specific elasticities; and
- \(\varepsilon>0\): a small stabilizing constant.

Explain the model in plain language:

> Realized output is the lesser of what the technology can reliably produce and what the relevant market, institution, or mission can absorb, after applying the
↪  remaining real-world bottlenecks.

Instantiate both branches numerically or symbolically for the target. Show which observations identify each term and which terms remain assumed.

Propagate uncertainty through both branches and through branch-specific and transfer bottlenecks. Where direct estimates are unavailable, state transparent distributions
↪  or sensitivity ranges for market size, adoption, utilization, technical constraints, demand constraints, and transfer constraints. Preserve dependence where evidence
↪  supports it; otherwise disclose the independence approximation. Never report a narrow realized-output interval produced only by holding the active capacity or demand
↪  ceiling fixed.

Prevent double-counting. A constraint already represented inside technical output or absorptive demand must not also be included unchanged in \(B_s(t)\).

Where several bottlenecks jointly matter, use the generalized bottleneck operator:

\[
B_\rho(t)=
\left[
\sum_{k=1}^{K}w_k b_k(t)^{-\rho}
\right]^{-1/\rho},
\]

with \(b_k(t)\in(0,1]\), \(\sum_k w_k=1\), and \(\rho>0\). As \(\rho\to\infty\), the operator approaches a strict weakest-link bottleneck.

Run a double-counting audit and list every factor, the branch in which it appears, and why it appears only there.

### Generalized realization family

Retain the compact hard-min equation as the primary, interpretable model. Where branch-specific constraints or partial substitution are materially plausible, also test
↪  the generalized family:

\[
\boxed{
Y_s(t)=B_s^{X}(t)
\left[
\alpha_s\bigl(B_s^{Q}(t)Q_s(t)\bigr)^{-\rho_s}
+
(1-\alpha_s)\bigl(B_s^{Z}(t)Z_s(t)\bigr)^{-\rho_s}
\right]^{-1/\rho_s}
}
\]

where \(Z_s(t)=M_s(t)D_s(t)U_s(t)\), \(B_s^Q\) constrains technical production, \(B_s^Z\) constrains absorptive demand, and \(B_s^X\) constrains transfer into accepted
↪  realization. As \(\rho_s\to\infty\), the family approaches the branch-specific hard minimum

\[
Y_s(t)=B_s^X(t)\min\{B_s^Q(t)Q_s(t),B_s^Z(t)Z_s(t)\}.
\]

Use the generalized family only as a disclosed sensitivity model. Report \(\alpha_s\), \(\rho_s\), branch assignments, and the forecast difference from the compact
↪  canonical case. Do not allow the extension to hide unidentified assumptions.

---

# VI. MEASURE LATENT CAPABILITY WITHOUT BENCHMARK SATURATION

When raw benchmark percentages saturate or task sets differ in difficulty, prefer a latent-variable model such as:
```

```
\[
\Pr(\text{success on task }i\mid t)=
\sigma\!\left[a_i(\theta(t)-b_i)\right],
\qquad
\sigma(x)=\frac{1}{1+e^{-x}},
\]

where \(b_i\) is task difficulty, \(a_i\) is discrimination, and \(\theta(t)\) is the latent capability frontier.

For a target task \(j\):

\[
p_j(t)=\sigma\!\left[\alpha_j(\theta(t)-d_j)\right].
\]

Do not equate a benchmark gain with an economic gain unless the benchmark change moves task success through a decision-relevant reliability threshold under comparable
↪  conditions.

---

# VII. FIT COMPETING MATHEMATICAL REGIMES

Do not assume a model class before examining the data. Fit or seriously evaluate at least:

### 1. Linear

\[
X_L(t)=X_0+vt.
\]

### 2. Exponential

\[
X_E(t)=X_0e^{gt}.
\]

### 3. Accelerating exponential

\[
X_A(t)=X_0\exp\left(g_0t+\tfrac12at^2\right).
\]

### 4. Accelerating model with decaying acceleration

\[
\dot g(t)=a_0e^{-\kappa t},
\]

\[
g(t)=g_0+\frac{a_0}{\kappa}(1-e^{-\kappa t}),
\]

\[
X(t)=X_0\exp\left[
 g_0t+\frac{a_0}{\kappa}t-
 \frac{a_0}{\kappa^2}(1-e^{-\kappa t})
\right].
\]

### 5. Constrained logistic or generalized logistic

\[
\dot X=rX\left[1-\left(\frac{X}{K}\right)^\nu\right].
\]

For \(\nu=1\):

\[
X(t)=\frac{K}{1+\left(\frac{K}{X_0}-1\right)e^{-rt}}.
\]

### 6. Moving-ceiling constrained model

\[
\dot X=g(t)XB(t)
\left[1-\left(\frac{X}{K(t)}\right)^\nu\right].
\]

### 7. Regime-switching or change-point model

Permit a transition time \(\tau\), different pre- and post-transition parameters, and uncertainty over \(\tau\).

Use out-of-sample performance wherever the data permit. Prefer:

- rolling-origin backtesting;
- historical pseudo-forecasts using only information available at each origin;
- leave-one-period-out validation;
- predictive log likelihood;
- continuous ranked probability score where predictive distributions are available;
- interval coverage;
- AICc or BIC as secondary checks.

Do not select a model solely because it most closely fits all historical observations in-sample.

For model \(m\), calculate:
```

$$
\mathrm{AICc}_m=2k_m-2\ln\widehat L_m+
\frac{2k_m(k_m+1)}{n-k_m-1}.
$$

Akaike weights:

$$
w_m=\frac{e^{-\Delta_m/2}}{\sum_j e^{-\Delta_j/2}}.
$$

When the sample is too small for meaningful discrimination:

- state that limitation;
- use model averaging;
- widen uncertainty ranges;
- emphasize trigger-based updating;
- avoid spurious parameter precision.

Show a compact model-comparison table containing model, fitted parameters, historical fit, backtest performance, strengths, failure modes, and scenario weight.

VIII. MODEL A SERIOUS DISCONTINUITY SCENARIO

The discontinuity scenario must be analytically serious, conditional, and uncertain—not decorative.

Model a structural change in which AI increasingly contributes to:

- AI research;
- coding;
- experiment design;
- evaluation;
- debugging;
- model optimization;
- infrastructure operation;
- deployment; and
- successor-system improvement.

Let $A_R(t)\in[0,1]$ be the independently verified and authorized fraction of the improvement loop that can operate autonomously.

Use:

$$
\dot g(t)=\left[\kappa A_R(t)-\delta\right]g(t).
$$

If $q=\kappa A_R-\delta>0$, then:

$$
g(t)=g_\tau e^{q(t-\tau)},
$$

and:

$$
\boxed{
X_D(t)=JX(\tau^-)
\exp\left[
\frac{g_\tau}{q}\left(e^{q(t-\tau)}-1\right)
\right]
}
$$

Define:

- τ: regime-change time;
- $J\ge1$: immediate capability or productivity jump;
- g_τ: growth rate at transition;
- q: rate at which the growth rate itself compounds;
- A_R: autonomous recursive-improvement share; and
- δ: friction, failed experiments, depreciation, verification cost, and coordination loss.

Apply physical and institutional constraints:

$$
\dot X=g(t)XB_R(t)
\left[1-\left(\frac{X}{K(t)}\right)^\nu\right].
$$

Do not interpret an unconstrained double-exponential or finite-time expression as literal infinite physical output.

Where evidence permits, model transition timing with a hazard function:

$$
\lambda(t)=\lambda_0\exp\left[\gamma^\top z(t)\right],
$$

$$
P(\tau\le T)=1-\exp\left[-\int_0^T\lambda(u)\,du\right].
$$

Potential trigger variables $z(t)$ include:

- independently reproduced autonomous AI-research gains;

- verified multi-generation improvement under equal constraints;
- major experiment-loop compression;
- long-horizon autonomous reliability;
- automated evaluation and debugging;
- compute and energy availability;
- deployment authority; and
- transfer of improvements beyond the training or evaluation environment.

Do not assign precise probabilities unsupported by evidence. Use explicit ranges, conditional probabilities, or qualitative probability bands where appropriate.

IX. RECOMPUTE EXECUTION FROM FIRST PRINCIPLES

Do not rely only on historical project timelines. Decompose the target into a directed acyclic task graph $G=(V,E)$ where possible.

For each task i, estimate:

- autonomous success probability p_i;
- maximum autonomous attempts m_i;
- AI execution time τ_i^{AI};
- verification time τ_i^V;
- human fallback time τ_i^H;
- approval or authority latency $\ell_i^{authority}$;
- external waiting time $\ell_i^{external}$;
- task cost;
- verified reliability;
- dependencies;
- parallelizability; and
- correlated-failure group.

Expected attempts:

$$
N_i^{attempt}=\frac{1-(1-p_i)^{m_i}}{p_i}.
$$

Escalation probability:

$$
P_i^{escalation}=(1-p_i)^{m_i}.
$$

Expected task duration:

$$
\bar\tau_i=
N_i^{attempt}(\tau_i^{AI}+\tau_i^V)
+(1-p_i)^{m_i}\tau_i^H
+\ell_i^{authority}
+\ell_i^{external}.
$$

Total work:

$$
W=\sum_i\bar\tau_i.
$$

Critical path:

$$
CP=\max_{\pi\in\mathcal P(G)}\sum_{i\in\pi}\bar\tau_i.
$$

With n parallel workers or agents and parallel-efficiency exponent $0<\eta\le1$:

$$
n_{eff}=n^\eta.
$$

Estimated workflow completion time:

$$
\boxed{
T_{workflow}\approx
\max\left\{CP,\frac{W}{n^\eta}\right\}
+H_{coord}(n)
}
$$

Per-task reliability after attempts:

$$
r_i^*=1-(1-r_i)^{m_i}.
$$

Workflow reliability with correlated-failure penalty Ω_{corr}:

$$
\ln R_{workflow}=\sum_i\ln r_i^*-\Omega_{corr}.
$$

Expected task cost:

$$

```
C_i=N_i^{attempt}(C_i^{AI}+C_i^V)+(1-p_i)^{m_i}C_i^H.
\]

Approximate cost per verified successful workflow:

\[
C_{verified}=\frac{\sum_i C_i}{R_{workflow}}.
\]

Produce a practical table showing:

- what can now occur in minutes;
- what can now occur in hours;
- what can now occur in days;
- what can run autonomously;
- what can run in parallel;
- what requires verification;
- what requires human authority; and
- what remains physically or institutionally rate-limited.

Do not let parallelism erase the critical path, approvals, external waiting, coordination overhead, or correlated failures.

---

# X. MODEL DEMAND, ADOPTION, AND ABSORPTIVE CAPACITY

For addressable population or mission volume \(M(t)\), adoption fraction \(D(t)\), and average utilization \(U(t)\):

\[
Q^{demand}(t)=M(t)D(t)U(t).
\]

Use a diffusion model where appropriate:

\[
\dot D(t)=[p+qD(t)][1-D(t)],
\]

or a logistic model:

\[
\dot D=r_DD(1-D).
\]

Explicitly model sales cycles, procurement, integration, training, switching costs, trust, policy, and authority where they slow adoption.

Do not assume technical availability causes instantaneous adoption or utilization.

---

# XI. PRODUCE FOUR PRIMARY FORECASTS

Generate four distinct, auditable scenarios. The downside case is a constraint stress scenario rather than a claim that deterioration is inevitable.

## A. Base forecast

Assume:

- current verified progress continues;
- some saturation or friction occurs;
- adoption and institutions adjust gradually;
- no major recursive-improvement discontinuity occurs.

Use the best constrained or model-averaged specification.

## B. Accelerated forecast

Assume:

- recent measured acceleration is real;
- costs, time compression, automation, reliability, and parallelism continue improving;
- acceleration may decay unless evidence supports persistence;
- adoption and infrastructure respond faster than in the base case.

## C. Downside or constraint-stress forecast

Assume one or more major constraints release more slowly, reverse, or become binding. Model the mechanism explicitly: reliability plateau, evaluation failure, adoption
↪  rejection, authority restriction, regulation, security incident, compute or energy shock, capital withdrawal, supply-chain delay, or physical bottleneck. Do not
↪  create the downside case by applying an unexplained arbitrary haircut.

## D. Discontinuous forecast

Assume:

- a structural regime transition occurs;
- a meaningful portion of the improvement loop becomes independently verified and autonomously executable;
- the growth rate itself begins compounding;
- realized output remains constrained by authority, trust, regulation, compute, energy, capital, supply chains, security, and physical execution.

For each scenario report:

- governing equation;
- central estimate;
- 50%, 80%, and 95% intervals where defensible;
- scenario probability or probability range;
- assumptions;
```

```
- forecast values at decision-relevant dates;
- milestone dates;
- leading indicators;
- invalidation conditions;
- dominant bottleneck;
- principal upside and downside.

Also include a downside or constraint stress test where material.

Do not combine the scenarios into a single expected value without also reporting the scenario-specific distributions.

---

# XII. CALCULATE MILESTONE DATES

For milestone level \(M\), define:

\[
T_M=\inf\left\{t:
Y(t)\ge M,\;
R(t)\ge R_M,\;
B_k(t)\ge B_{k,M}
\right\}.
\]

A milestone is not achieved merely because a raw capability benchmark crosses a threshold. It must also satisfy the required reliability, verification, authority,
↪  adoption, deployment, economic, or physical realization conditions.

Useful closed forms include:

### Exponential crossing

\[
T_M=\frac{\ln(M/X_0)}{g}.
\]

### Accelerating-exponential crossing

\[
T_M=\frac{-g_0+\sqrt{g_0^2+2a\ln(M/X_0)}}{a}.
\]

### Logistic crossing

\[
T_M=\frac{1}{r}\ln\left[
\frac{K/X_0-1}{K/M-1}
\right].
\]

### Double-exponential crossing after transition

\[
T_M=\tau+\frac1q\ln\left[
1+\frac q{g_\tau}\ln\left(
\frac{M}{JX(\tau^-)}
\right)
\right].
\]

Report milestone dates as ranges or probability distributions when uncertainty is material.

---

# XIII. SPECIALIZATION FOR REVENUE OR SALES TARGETS

When the target concerns sales, bookings, recurring revenue, or cash collections, model separately:

- qualified opportunity volume \(L(t)\);
- conversion probability \(q(t)\);
- sales-cycle lag \(\ell(t)\);
- average contract value \(V(t)\);
- delivery capacity \(K_{delivery}(t)\);
- implementation capacity;
- expansion;
- churn;
- collection risk; and
- concentration risk.

Expected closed deals:

\[
N_{closed}(t)=L(t-\ell)q(t-\ell).
\]

Expected bookings:

\[
Bookings(t)=V(t)\min\left\{
L(t-\ell)q(t-\ell),K_{delivery}(t)
\right\}.
\]

For identifiable opportunities:

\[
```

```
\mathbb E[Bookings]=\sum_i p_iV_i.
\]

Variance, including dependence where material:

\[
\operatorname{Var}(Bookings)=
\sum_i p_i(1-p_i)V_i^2
+2\sum_{i<j}\operatorname{Cov}(I_iV_i,I_jV_j).
\]

For recurring revenue:

\[
\frac{dMRR}{dt}=MRR_{new}+MRR_{expansion}-MRR_{churn}.
\]

Do not infer revenue directly from technical capability. Show exactly how capability, cost, time compression, automation, reliability, trust, and adoption affect
↪  opportunity volume, conversion, sales cycle, pricing, delivery capacity, retention, and collections.

---

# XIV. IDENTIFY AND QUANTIFY BOTTLENECKS

For each bottleneck, report:

- current severity;
- evidence grade;
- mechanism of impact;
- whether it limits capability, verification, deployment, adoption, or realization;
- estimated delay or output penalty;
- whether it is parallelizable;
- earliest plausible release condition;
- leading indicator;
- mitigation;
- residual risk.

Cover, where relevant:

- authority;
- trust;
- verification;
- organizational adoption;
- legal and regulatory constraints;
- security;
- capital;
- compute;
- energy;
- data;
- supply chains;
- physical construction;
- customer acquisition;
- procurement;
- sales cycles;
- human coordination; and
- institutional legitimacy.

Rank bottlenecks by marginal effect on \(Y(t)\), not merely by narrative prominence.

---

# XV. BUILD A TRIGGER DASHBOARD

For each trigger, specify:

- metric;
- current value;
- threshold;
- source;
- evidence grade;
- why it matters;
- affected scenario or parameter;
- forecast change if crossed;
- recalculation action.

Include acceleration, deceleration, and regime-change triggers.

Potential acceleration triggers include:

- large verified gain on fresh economically meaningful tasks;
- collapse in cost per verified outcome;
- major increase in autonomous operating horizon;
- independently reproduced automated-research gains;
- sharp sales-cycle compression;
- rapid institutional adoption;
- material compute or energy expansion.

Potential deceleration triggers include:

- reliability plateaus;
- benchmark contamination or saturation;
- rising verification cost;
- regulatory restriction;
- compute or energy bottlenecks;
- weak conversion despite capability gains;
- organizational rejection;
```

```
- security incidents;
- supply-chain delays.

---

# XVI. QUANTIFY UNCERTAINTY AND SENSITIVITY

Separate:

1. measurement uncertainty;
2. parameter uncertainty;
3. model uncertainty;
4. scenario uncertainty;
5. discontinuity-timing uncertainty;
6. execution uncertainty; and
7. deep structural uncertainty.

Use model averaging where appropriate:

\[
p(Y_T\mid\mathcal D)=
\sum_s\pi_s\sum_m w_{m,s}
 p(Y_T\mid M_{m,s},\mathcal D).
\]

Decompose predictive variance:

\[
\operatorname{Var}(Y)=
\mathbb E_M[\operatorname{Var}(Y\mid M)]
+\operatorname{Var}_M[\mathbb E(Y\mid M)].
\]

Report, where defensible:

- median;
- 50% interval;
- 80% interval;
- 95% outer range;
- milestone-crossing probability;
- scenario probabilities or ranges.

Sensitivity to parameter \(\Theta_j\):

\[
S_j=\frac{\partial\ln Y}{\partial\ln\Theta_j}.
\]

Show which assumptions dominate the result. Do not conceal uncertainty inside a single expected value or a falsely precise date.

---

# XVII. DETERMINE THE OPTIMAL PLAN FOR TODAY'S REGIME

Do not merely forecast. Convert the forecast into action.

For candidate action \(a\), use:

\[
\boxed{
a^*=\arg\max_a\left[
\sum_s\pi_s\left(
NPV_s(a)+V_{learning,s}(a)+V_{option,s}(a)
\right)
-\lambda\,CVaR_\alpha(L_s(a))
\right]
}
\]

subject to budget, authority, reliability, security, legal constraints, and maximum irreversible loss.

When scenario probabilities are too uncertain, use minimax regret:

\[
a^*=\arg\min_a\max_s\left[V_s^*-V_s(a)\right].
\]

Recommend:

1. no-regret actions that perform well across all scenarios;
2. accelerated-regime actions that should begin now;
3. discontinuity-readiness options that preserve upside;
4. actions that should remain reversible;
5. actions that should be delayed pending evidence;
6. stop, expand, and pivot triggers; and
7. the next highest-information experiment.

Favor plans that generate evidence quickly, shorten decision cycles, preserve optionality, scale in parallel, and avoid irreversible commitments based on fragile
↪  forecasts.

---

# XVIII. SET A DATED RECALCULATION POINT

Set an exact scheduled recalculation date.
```

```
Default:

\[
\Delta=\min\{30\text{ days},0.1H\},
\]

where \(H\) is the forecast horizon.

Define:

\[
t_{recalc}=\min\left\{
 t_0+\Delta,
 T_{forecast\ break},
 T_{trigger},
 T_{regime\ change}
\right\}.
\]

Recalculate earlier when:

\[
\left|
\ln Y_{observed}(t)-\ln\widehat Y(t)
\right|>
 z_\alpha\sigma_{pred}(t),
\]

or when a material trigger threshold is crossed.

State:

- scheduled recalculation date;
- events that force immediate recalculation;
- data that must be collected before that date;
- owner of each measurement where an operating plan is requested.

---

# XIX. PRODUCTION CONTRACT AND PROSPECTIVE REGISTRY

For an operational forecast, emit a machine-readable record conforming to the versioned input and output schemas. The record must contain:

- protocol and engine versions;
- a unique forecast identifier;
- exact research cutoff and generation time;
- target definition, unit, scope, and success criterion;
- dated comparable observations and evidence identifiers;
- source-quality and comparability metadata;
- fitted models, parameters, likelihood or scoring values, and model weights;
- scenario probabilities, assumptions, parameters, and invalidation conditions;
- median, mean, 50%, 80%, and 95% intervals at each reporting date;
- milestone-crossing probabilities and date distributions;
- bottlenecks, triggers, actions, and recalculation gate;
- input, code, environment, and output hashes.

Before an externally consequential forecast is acted upon, validate the input and output against the schemas. Store the forecast in an append-only, hash-chained
↪  prospective registry before outcomes are known. The registry record must include the previous-record hash, current-record hash, input hash, forecast hash, and
↪  registration timestamp. Never rewrite a registered forecast; append a correction record that links to the superseded record.

Evaluate matured forecasts with proper scores. At minimum, report log error for positive continuous targets, interval coverage, continuous ranked probability score or an
↪  equivalent proper score, weighted interval score, and Brier or log score for threshold events. Separate retrospective demonstrations from genuinely prospective
↪  forecasts.

---

# XX. REPRODUCIBILITY AND AUDIT TRAIL

Provide enough information for an independent analyst to reproduce the forecast:

- versioned prompt and model specification;
- exact source ledger;
- data snapshot or retrieval script;
- transformation log;
- units and comparability bridges;
- fitted parameters and constraints;
- random seed;
- software versions;
- model-selection scores;
- scenario assumptions;
- generated outputs and checksums.

When source terms prohibit redistribution, provide a retrieval adapter and checksum rather than republishing restricted data.

Label any synthetic or illustrative dataset unambiguously. Never present simulated results as empirical findings.

---

# XXI. REQUIRED OUTPUT ORDER

Return the analysis in exactly this order:

1. **Executive Forecast**
   - one-paragraph conclusion;
   - base, accelerated, downside, and discontinuous headline outcomes;
   - immediate decision implication.
```

```
2. **Scope, Definitions, and Research Cutoff**
   - target definition;
   - baseline;
   - unit;
   - exact evidence cutoff;
   - material assumptions.

3. **Newest Evidence**
   - source-quality table;
   - current, 6-month, and 12-month comparisons.

4. **Current Pace and Calculations**
   - capability;
   - cost decline;
   - task-time compression;
   - automation;
   - reliability;
   - adoption;
   - measured acceleration or deceleration.

5. **Compact Canonical Model**
   - display both boxed equations;
   - define every variable;
   - explain the model in plain language;
   - instantiate it for the target;
   - show the double-counting audit.

6. **Mathematical Model Fit**
   - candidate models;
   - fitted parameters;
   - backtests;
   - selected or averaged model;
   - model limitations.

7. **First-Principles Capacity Recalculation**
   - minutes, hours, and days;
   - autonomy;
   - parallelism;
   - verification;
   - authority;
   - critical path;
   - cost per verified outcome.

8. **Forecast Scenarios**
   - base;
   - accelerated;
   - downside stress;
   - discontinuous;
   - uncertainty ranges;
   - probabilities or probability ranges.

9. **Milestones and Dates**
   - threshold;
   - base date;
   - accelerated date;
   - discontinuous date;
   - confidence;
   - prerequisites.

10. **Bottlenecks**
    - severity;
    - marginal impact;
    - release condition;
    - mitigation.

11. **Trigger Dashboard**
    - acceleration triggers;
    - deceleration triggers;
    - regime-change triggers.

12. **Optimal Plan for Today's Regime**
    - no-regret actions;
    - option-preserving actions;
    - next experiment;
    - stop, expand, and pivot criteria.

13. **Confidence, Sensitivities, and Assumptions**
    - observed versus derived versus estimated versus assumed;
    - dominant sensitivities;
    - key unknowns.

14. **Dated Recalculation Point**
    - exact date;
    - required evidence;
    - early-recalculation triggers.

15. **Sources, Data Provenance, and Calculation Notes**
    - source ledger;
    - data and code versions;
    - reproducibility notes.

16. **Machine-Readable Forecast Record**
    - schema and engine versions;
    - forecast identifier;
    - input and output SHA-256 digests;
```

```
    - prospective-registry status;
    - exact path or artifact identifier for the validated JSON record.

---

# XXII. QUALITY-CONTROL CHECKLIST

Before returning the answer, verify that:

- the newest reliable evidence was used;
- exact dates and time zones are shown;
- event, measurement, release, publication, and retrieval dates are distinguished;
- observed, derived, estimated, assumed, and scenario-conditional values are separated;
- incompatible metrics were not averaged or silently joined;
- capability gains were not automatically translated into reliability, adoption, revenue, or physical output;
- raw inference cost was not substituted for cost per verified outcome;
- retries, verification, authority, external waiting, and human fallback were included;
- parallelism did not erase the critical path or coordination overhead;
- correlated failures were considered;
- adoption and bottleneck effects were not double-counted;
- benchmark saturation, contamination, and scaffold changes were considered;
- saturation and moving ceilings were considered;
- acceleration was measured rather than merely asserted;
- the discontinuity scenario changes system dynamics and is explicitly uncertain;
- the forecast can change regime during the horizon;
- out-of-sample or pseudo-out-of-sample evaluation was used where possible;
- uncertainty intervals widen where evidence is weak;
- the compact canonical model is visibly displayed, defined, explained, and instantiated;
- every major numerical claim is traceable to evidence or an explicit assumption;
- the input and output satisfy the versioned machine-readable schemas;
- any prospective forecast is registered before outcomes are observed and scored after maturation;
- the answer ends with a specific action plan and exact recalculation date;
- simulated or illustrative evidence is never presented as observed fact.

---

## FINAL DIRECTIVE

Never extrapolate stale timelines.

Forecast a world that may accelerate during the forecast itself.
```

B Selected Derivations

B.1 Technical capacity from an instantaneous growth rate

By definition $g_Q(t) = \mathrm{d}\ln Q(t)/\mathrm{d}t$. Integrating from 0 to t gives

$$\ln Q(t) - \ln Q_0 = \int_0^t g_Q(u)\,\mathrm{d}u,$$

so

$$Q(t) = Q_0 \exp\left[\int_0^t g_Q(u)\,\mathrm{d}u\right].$$

B.2 Technical-growth decomposition

Suppose

$$Q = Q_0 e^{\beta_\theta(\theta-\theta_0)} c^{-\beta_c} \tau^{-\beta_\tau} (A+\varepsilon)^{\beta_A} P^{\beta_P} R^{\beta_R}$$

up to normalization constants. Taking a time derivative of $\ln Q$ yields Equation (12).

B.3 Expected truncated geometric attempts

For success probability p and maximum m attempts, the number of attempts is at least k with probability $(1-p)^{k-1}$. Therefore

$$\mathbb{E}[N] = \sum_{k=1}^{m} \Pr(N \geq k) = \sum_{k=1}^{m} (1-p)^{k-1} = \frac{1-(1-p)^m}{p}.$$

B.4 Accelerating-exponential crossing time

Set $M = X_0 e^{g_0 t + at^2/2}$. Then

$$\tfrac{1}{2}at^2 + g_0 t - \ln(M/X_0) = 0.$$

For $a > 0$, the nonnegative root is

$$T_M = \frac{-g_0 + \sqrt{g_0^2 + 2a\ln(M/X_0)}}{a}.$$

B.5 Exponential and logistic crossing times

For $X = X_0 e^{gt}$,

$$T_M = \frac{\ln(M/X_0)}{g}.$$

For $X = K/[1 + (K/X_0 - 1)e^{-rt}]$,

$$T_M = \frac{1}{r}\ln\left(\frac{K/X_0 - 1}{K/M - 1}\right), \qquad 0 < M < K.$$

B.6 Double-exponential crossing time

From Equation (24), for $M > JX(\tau^-)$,

$$T_M = \tau + \frac{1}{q}\ln\left[1 + \frac{q}{g_\tau}\ln\left(\frac{M}{JX(\tau^-)}\right)\right].$$

B.7 Discontinuity hazard

For a nonnegative time-varying hazard $\lambda(t)$, survival is

$$S(T) = \exp\left[-\int_0^T \lambda(u)\,\mathrm{d}u\right],$$

so cumulative transition probability is $1 - S(T)$.

B.8 Hard-min limit of the generalized realization family

Let $x = B^Q Q > 0$, $z = B^Z Z > 0$, and

$$G_\rho(x,z) = \left[\alpha x^{-\rho} + (1-\alpha)z^{-\rho}\right]^{-1/\rho}.$$

Assume without loss of generality that $x \le z$. Then

$$G_\rho(x,z) = x\left[\alpha + (1-\alpha)(x/z)^\rho\right]^{-1/\rho}.$$

Since $(x/z)^\rho \to 0$ and $\alpha^{-1/\rho} \to 1$ as $\rho \to \infty$, $G_\rho(x,z) \to x = \min\{x,z\}$. Multiplying by B^X yields the hard-min limit stated after Equation (10).

B.9 Generalized bottleneck limit

Equation (14) is a weighted power mean of order $-\varrho$. As $\varrho \to \infty$, the order tends to $-\infty$, whose limit is the minimum component.

B.10 Workflow reliability

Independent retry success is one minus the probability all attempts fail: $r_i^* = 1-(1-r_i)^{m_i}$. Independent required tasks multiply. Taking logs converts the product to a sum; Ω_{corr} subtracts a common-mode penalty.

B.11 Uncertainty decomposition

The law of total variance gives

$$\mathrm{Var}(Y) = \mathbb{E}[\mathrm{Var}(Y \mid M)] + \mathrm{Var}(\mathbb{E}[Y \mid M]),$$

which separates within-model from between-model uncertainty.

C Variable Dictionary

Symbol	Meaning	Domain / unit
$Y_s(t)$	Realized target outcome under scenario s	Target-specific, nonnegative
Q_0	Baseline technically deliverable output	Positive
$g_{Q,s}(t)$	Instantaneous proportional growth rate of technical output	Inverse time
$M_s(t)$	Addressable market, mission volume, or demand pool	Nonnegative
$D_s(t)$	Adoption or penetration fraction	$[0,1]$
$U_s(t)$	Utilization or realized value per adopter	Nonnegative
$B_s(t)$	Compact residual bottleneck factor after double-counting audit	$(0,1]$
$B_s^Q(t)$	Technical-production bottleneck in generalized realization family	$(0,1]$
$B_s^Z(t)$	Absorptive-demand bottleneck in generalized realization family	$(0,1]$
$B_s^X(t)$	Transfer or realization bottleneck in generalized realization family	$(0,1]$
$Z_s(t)$	Absorptive demand, $M_s D_s U_s$	Same unit as Q_s
α_s	Branch weight in generalized realization family	$(0,1)$
ρ_s	Substitution/weakest-link sharpness in generalized realization family	Positive
$\theta(t)$	Latent capability level	Scale-dependent
$c(t)$	Cost per attempt or unit of work	Positive currency/unit
$\tau(t)$	Task-completion time	Positive time
$A(t)$	Automatable workflow share	$[0,1]$

Symbol	Meaning	Domain / unit
$P(t)$	Effective parallelism	Positive
$R(t)$	Verified reliability	$(0, 1]$
β_j	Target-specific elasticity of component j	Dimensionless
ε	Stabilizing constant for automation share	Positive, small
$b_k(t)$	Constraint factor k	$(0, 1]$
w_k	Bottleneck weight	Nonnegative, sums to 1
ϱ	Constraint-aggregator sharpness in Equation (14)	Positive
p_i	Per-attempt autonomous success probability for task i	$[0, 1]$
m_i	Maximum autonomous attempts before escalation	Positive integer
$\bar{\tau}_i$	Expected duration of task i	Time
W	Sum of expected task work	Time
CP	Expected critical-path duration	Time
n	Parallel worker or agent count	Positive integer
η	Parallel-efficiency exponent	$(0, 1]$
H_{coord}	Coordination overhead	Time
Ω_{corr}	Correlated-failure penalty in log reliability	Nonnegative
$A_R(t)$	Verified and authorized autonomous share of improvement loop	$[0, 1]$
τ	Regime-transition time (context-dependent; distinct from task time by subscript)	Time
J	Immediate multiplicative jump at transition	≥ 1
q	Post-transition compounding rate of the growth rate	Inverse time
$\lambda(t)$	Transition hazard	Inverse time
$K(t)$	Current regime-dependent ceiling	Positive
ν	Generalized logistic shape	Positive
π_s	Scenario probability or decision weight	$[0, 1]$, sums to 1 when probabilistic

D Reproducibility and Machine-Readable Protocol

The canonical repository destination is

`https://github.com/MontrealAI/forecasting-a-world-that-accelerates`.

The URL is a publication target until the repository is actually created and visible. The release never treats a local folder, checksum, or intended URL as an external timestamp.

D.1 Package map

Path	Purpose
`paper/`	LaTeX source, bibliography, generated tables and figures, and compiled preprint.
`protocol/prompt.*`	Complete versioned executable protocol reproduced in Appendix A.
`protocol/v2/`	Strict version 2 input, output, and registry JSON Schemas plus a complete reference case.
`protocol/variables.yaml`	Canonical variable dictionary and units.

`src/fwta/`	Python implementation of canonical equations, generalized realization, growth models, probabilistic ensembles, scenarios, task graphs, scoring, reporting, registry, and provenance.
`tests/`	Deterministic unit, schema, engine, registry, scoring, workflow, and reproduction tests.
`data/public/`	Frozen source-linked excerpts and controlled fixtures.
`data/provenance/`	Source, retrieval, selection, hash, and limitation records.
`results/ceiling/`	Complete reference forecast JSON and self-contained HTML report.
`results/empirical/`	METR and SWE-bench demonstration outputs.
`registry/`	Hash-chained prospective records, preregistration template, and verification instructions.
`release/`	SBOM, manifest, publication status, release notes, and validation report.
`scripts/`	Build, verification, packaging, repository configuration, and release tools.
`.github/`	CI, compatibility matrix, code scanning, Pages, and draft-release workflows.

D.2 One-command forecast

After installing the package, a complete reference run is:

```
fwta forecast protocol/v2/examples/canonical-reference-input.yaml \
  --schema protocol/v2/input.schema.json \
  --output results/ceiling/reference-forecast.json \
  --report results/ceiling/reference-forecast.html \
  --output-schema protocol/v2/output.schema.json
```

The report is standalone and includes the canonical equations, scenario chart, milestones, bottlenecks, triggers, actions, limitations, and embedded machine-readable output.

D.3 Clean local reproduction

```
python3 -m venv .venv
source .venv/bin/activate
python -m pip install --upgrade pip
python -m pip install -e '.[dev]'
make verify
```

The default tests, controlled experiments, frozen excerpts, and reference forecast require no API key and make no network request. Dependency installation may require network access. The package also supplies a Dockerfile, Compose definition, pinned direct requirements, SBOM generator, Windows instructions, and CI compatibility matrix.

D.4 Prospective registration

A completed forecast can be appended to the local registry and verified with:

```
fwta register results/ceiling/reference-forecast.json \
  --registry registry/records.json \
  --record results/ceiling/registry-record.json
fwta registry-verify --registry registry/records.json
```

The chain detects mutation within the published sequence. Independent dating requires a public signed release, archival deposit, or preprint service.

D.5 Release integrity and boundaries

The release process builds source and wheel distributions, recompiles the paper, validates schemas, reruns experiments, generates a CycloneDX SBOM, scans for common secret patterns, and emits SHA-256 manifests. A checksum proves byte identity with a specified artifact; it does not independently prove authorship, legal priority, non-infringement, or publication time. The publication workflow creates a draft release so an authorized owner can complete the IP and legal gate before public disclosure.

E Governance, Use, and Legal Boundaries

E.1 License map

Software is offered under AGPL-3.0-or-later. The paper, prompt, schemas, documentation, and original research artifacts are offered under CC BY-NC-SA 4.0. Third-party material is not relicensed. No trademark license is granted. Separate commercial rights require a signed agreement with MONTREAL.AI. The controlling path-specific map is in `LICENSE.md`.

E.2 No warranty or professional opinion

The framework is research software and methodology. Forecasts are conditional distributions, not promises. They depend on target definition, evidence quality, model choice, parameter estimates, revisions, implementation, and rare events. Users remain responsible for independent verification, professional review where appropriate, authority, data rights, and compliance with applicable law.

E.3 Contribution controls

External contributions require DCO sign-off and the project contributor license agreement before merge. The policy preserves provenance, public licensing, coherent commercial licensing, and maintainer control. Contribution does not confer ownership, equity, governance, partnership, agency, employment, or trademark rights.

E.4 Publication record

Version 2.0.0 is prepared for the exact standalone repository specified above. At package assembly, the repository and external preprint record had not yet been created. The release documentation prohibits claiming a DOI, cryptographic signature, or public timestamp until the corresponding public record exists.

References

Hirotugu Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19 (6):716–723, 1974. doi:10.1109/TAC.1974.1100705.

Gene M. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In *Proceedings of the April 18–20, 1967, Spring Joint Computer Conference*, pages 483–485, 1967. doi:10.1145/1465482.1465560.

Frank M. Bass. A new product growth for model consumer durables. *Management Science*, 15(5):215–227, 1969. doi:10.1287/mnsc.15.5.215.

Kathryn Bonney, Cory Breaux, Catherine Buffington, Emin Dinlersoz, Lucia Foster, Nathan Goldschlag, John Haltiwanger, Zachary Kroff, and Keith Savage. Tracking firm use of ai in real time: A snapshot from the business trends and outlook survey. Technical Report CES-WP-24-16, U.S. Census Bureau, Center for Economic Studies, March 2024. URL https://www.census.gov/library/working-papers/2024/adrm/CES-WP-24-16.html.

Kathryn Bonney, Cory Breaux, Emin Dinlersoz, Lucia Foster, John Haltiwanger, and Aditya Pande. The microstructure of ai diffusion: Evidence from firms, business functions, and worker tasks. Technical Report CES-WP-26-25, U.S. Census Bureau, Center for Economic Studies, April 2026. URL https://www.census.gov/library/working-papers/2026/adrm/CES-WP-26-25.html.

Kenneth P. Burnham and David R. Anderson. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer, New York, 2 edition, 2002. doi:10.1007/b97636.

Mingguang Chen, Licheng Wang, and Bo Qu. Recursive self-improvement in ai: From bounded self-refinement to autonomous research loops. *arXiv preprint arXiv:2607.07663*, 2026. doi:10.48550/arXiv.2607.07663.

Tom Davidson, Basil Halperin, Thomas Houlden, and Anton Korinek. When does automating ai research produce explosive growth? feedback loops in innovation networks. Working Paper 35155, National Bureau of Economic Research, April 2026.

Mert Demirer, Leon Musolff, and Liyuan Yang. Writing code vs. shipping code: Productivity effects across generations of ai coding tools. Working Paper 35275, National Bureau of Economic Research, May 2026.

Francis X. Diebold and Roberto S. Mariano. Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3):253–263, 1995. doi:10.1080/07350015.1995.10524599.

Ege Erdil, Andrei Potlogea, Tamay Besiroglu, Edu Roldan, Anson Ho, Jaime Sevilla, Matthew Barnett, Matej Vrzla, and Robert Sandler. GATE: An integrated assessment model for ai automation. *arXiv preprint arXiv:2503.04941*, 2025. doi:10.48550/arXiv.2503.04941.

Mehmet Hamza Erol, Batu El, Mirac Suzgun, Mert Yuksekgonul, and James Zou. Cost-of-pass: An economic framework for evaluating language models. *arXiv preprint arXiv:2504.13359*, 2025. doi:10.48550/arXiv.2504.13359.

GitHub. Octoverse: A new developer joins github every second as ai leads typescript to number one, 2025. URL https://github.blog/news-insights/octoverse/octoverse-a-new-developer-joins-github-every-second-as-ai-leads-typescript-to-1/. Official GitHub Octoverse report; accessed 2026-07-28.

GitHub. Copilot coding agent now starts work 50 percent faster, March 2026. URL https://github.blog/changelog/2026-03-19-copilot-coding-agent-now-starts-work-50-faster/. Official GitHub Changelog; accessed 2026-07-28.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi:10.1198/016214506000001437.

Adam Grundy, Cory Breaux, and Dhanapati Khatiwoda. Large firms with at least 20 employees biggest ai users. U.S. Census Bureau, May 2026. URL https://www.census.gov/library/stories/2026/05/ai-use-businesses.html. Reports 19.8 percent for the collection period ending May 3, 2026 and documents the November 2025 wording change; accessed 2026-07-28.

Hans Gundlach, Jayson Lynch, Matthias Mertens, and Neil Thompson. The price of progress: Algorithmic efficiency and the falling cost of ai inference. *arXiv preprint arXiv:2511.23455*, 2025. doi:10.48550/arXiv.2511.23455.

John L. Gustafson. Reevaluating amdahl's law. *Communications of the ACM*, 31(5):532–533, 1988. doi:10.1145/42411.42415.

James D. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2):357–384, 1989. doi:10.2307/1912559.

Rob J. Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, 3 edition, 2021. URL https://otexts.com/fpp3/.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations*, 2024. doi:10.48550/arXiv.2310.06770. Oral presentation.

Benjamin F. Jones. Artificial intelligence in research and development. Working Paper 34312, National Bureau of Economic Research, October 2025.

Charles I. Jones. R&d-based models of economic growth. *Journal of Political Economy*, 103(4):759–784, 1995. doi:10.1086/262002.

Charles I. Jones. A.i. and our economic future. Working Paper 34779, National Bureau of Economic Research, 2026. Revised June 2026.

Ezra Karger, Houtan Bastani, Yueh-Han Chen, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E. Tetlock. Forecastbench: A dynamic benchmark of ai forecasting capabilities. *arXiv preprint arXiv:2409.19839*, 2024. doi:10.48550/arXiv.2409.19839.

Michael Kremer. The o-ring theory of economic development. *Quarterly Journal of Economics*, 108(3):551–575, 1993. doi:10.2307/2118405.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney von Arx, Ryan Bloom, Thomas Broadley, Haoxing Du, Brian Goodrich, Nikola Jurkovic, Luke Harold Miles, Seraphina Nix, Tao Lin, Neev Parikh, David Rein, Lucas Jun Koba Sato, Hjalmar Wijk, Daniel M. Ziegler, Elizabeth Barnes, and Lawrence Chan. Measuring ai ability to complete long tasks. *arXiv preprint arXiv:2503.14499*, 2025. doi:10.48550/arXiv.2503.14499.

METR. Task-completion time horizons of frontier ai models, May 2026. URL https://metr.org/time-horizons/. Last updated May 8, 2026; accessed July 28, 2026.

William D. Nordhaus. Are we approaching an economic singularity? information technology and the future of economic growth. *American Economic Journal: Macroeconomics*, 13(1):299–332, 2021. doi:10.1257/mac.20170105.

Stephan Rabanser, Sayash Kapoor, Peter Kirgis, Kangheng Liu, Saiteja Utpala, and Arvind Narayanan. Towards a science of ai agent reliability. *arXiv preprint arXiv:2602.16666*, 2026. doi:10.48550/arXiv.2602.16666.

Georg Rasch. *Probabilistic Models for Some Intelligence and Attainment Tests.* University of Chicago Press, Chicago, 1980. Expanded edition of the 1960 work.

R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2(3): 21–41, 2000. doi:10.21314/JOR.2000.038.

Gopal P. Sarma, Sunny D. Bhatt, Michael Jacob, and Rachel Steratore. Artificial general intelligence forecasting and scenario analysis: State of the field, methodological gaps, and strategic implications. *arXiv preprint arXiv:2604.22766*, 2026. doi:10.48550/arXiv.2604.22766.

SWE-bench. SWE-bench verified leaderboard and evaluation repository, 2026. URL https://www.swebench.com/. Official benchmark and leaderboard materials; accessed 2026-07-28.

Philip Trammell and Anton Korinek. Economic growth under transformative ai. Working Paper 31815, National Bureau of Economic Research, 2023. Revised April 2026.

U.S. Census Bureau. Only 3.8% of businesses use ai to produce goods and services, highest use in information sector, November 2023. URL https://www.census.gov/library/stories/2023/11/businesses-use-ai.html. Reference period October 23–November 5, 2023; accessed 2026-07-28.

U.S. Census Bureau. July 2025: Science fiction, July 2025. URL https://www.census.gov/about/history/stories/monthly/2025/july-2025.html. Reports that the May 2025 BTOS estimate was close to 10 percent; accessed 2026-07-28.