LLMindex

A Psychometric, Contamination-Resistant, and Fully Auditable Methodology for Ranking Large Language Models

Methodology White Paper — Index Version 0.2.0

Simon-Pierre Boucher
contact@spboucher.ai
www.llmindex.io

August 2026

Abstract

Public leaderboards have become the de facto interface between frontier language-model development and its consumers, yet the dominant evaluation paradigm is failing in two structural ways: *saturation*, in which top models compress into statistically indistinguishable score bands on static benchmarks, and *contamination*, in which fixed test sets leak into training corpora and convert measurement into memorization. LLMindex is an operating evaluation platform built to address both failures at the level of measurement design rather than test-set curation. Every scored item is instantiated at evaluation time from seeded, versioned template generators, so no fixed test set exists to memorize; ability is estimated with a two-parameter logistic (2PL) item response model rather than raw accuracy, so uninformative items are automatically identified and retired; open-ended capabilities are ranked through pairwise duels judged by a cross-provider panel with position-swap bias controls and aggregated with a Bradley–Terry model; and answer stability, confidence calibration, and a directly measured memorization gap enter the score as explicit, separately reported sub-metrics. Cost and latency are published as a Pareto frontier and are never blended into quality. All scores carry 95% confidence intervals, every displayed number traces to an immutable, fully audited evaluation run, and the methodology itself is versioned under semantic versioning with a public changelog. This paper motivates the design, develops its mathematical foundations, explains how scores should be interpreted, situates the platform among existing evaluation efforts, and states its limitations and roadmap candidly.

Keywords: large language models, evaluation, item response theory, Bradley–Terry, benchmark contamination, calibration, LLM-as-judge, agentic evaluation

Contents

1 Introduction: Why Current Benchmarks Fall Short

Language-model benchmarks began as research instruments and became market infrastructure. Procurement decisions, deployment choices, and public narratives about model progress now route through a handful of numbers. Those numbers are increasingly unreliable, for reasons that are structural rather than incidental.

1.1 Saturation and the collapse of discrimination

A benchmark discriminates only where models can fail. On the static suites that dominate public discourse — MMLU [11], GSM8K, HumanEval and their descendants — frontier models now cluster above 90% accuracy. From a measurement-theoretic standpoint the consequence is precise: the Fisher information an item contributes to an ability estimate is proportional to $P(1-P)$, where P is the probability the examinee answers correctly (Section 5). An item every strong model solves contributes essentially *zero* information about how strong models differ. Empirically, sparse-benchmark analyses find that fewer than 3% of items on major benchmarks carry nearly all of their evaluative information [13], and adaptive-testing studies show that replacing raw accuracy with latent-ability estimation reorders a substantial fraction of model rankings [21]. Raw accuracy on a saturated suite is not a noisy version of the right answer; it is dominated by items that no longer measure anything.

1.2 Contamination converts measurement into memorization

Static test sets published on the public internet are, with high probability, present in the pretraining corpora of the models they are meant to evaluate [18]. The clearest evidence is perturbational: when the numeric values of GSM8K-style templates are re-randomized while the logical structure is preserved, measured accuracy drops materially and variance across instantiations inflates [17]. A model that solves the published instance but not its structural twin has memorized, not reasoned; a benchmark that cannot distinguish these two has lost its object of measurement.

1.3 Format compliance masquerades as capability

Evaluation harnesses grade what they can parse. Rigid answer-extraction rules — a single accepted tag, strict multiple-choice letter matching — systematically penalize model families whose reasoning-tuned outputs wrap answers in markdown, LaTeX `\boxed{}` expressions, or free-form prose. When the Open LLM Leaderboard replaced strict extraction with a lenient mathematical-equivalence grader, some model families' scores tripled and the top of the ranking reshuffled — with no change to any model. Regex-based extraction has been measured at only ∼74% accuracy [30]; formatting alone can move measured accuracy by tens of points [22]; and reasoning-tuned models comply with embedded format instructions less than 25% of the time [25, 32]. A harness that conflates parsing with ability measures its own regexes.

1.4 Single numbers hide the trade-offs that matter

A model that is slightly less accurate but five times cheaper, or equally accurate but incapable of saying *how sure it is*, is a materially different engineering proposition. Blending accuracy, price, speed, and reliability into one scalar destroys exactly the information a practitioner needs. Similarly, a point estimate without an interval invites over-reading: a three-point gap between

two models is meaningless if the measurement uncertainty is ten points, yet leaderboards routinely present such gaps as rank orderings.

1.5 Opacity prevents correction

Most public rankings cannot answer elementary audit questions: Which exact prompts were used? What did the model actually reply? Under which grader version? What changed between last month's score and today's? Without an immutable audit trail, errors persist silently and disputes cannot be adjudicated on evidence.

1.6 The LLMindex response

LLMindex responds to each failure at the level of design rather than patching:

1. **Against saturation**: items are *generated* with difficulty knobs calibrated so that frontier models fail a substantial fraction of them, and ability is estimated with a 2PL item response model in which uninformative items are automatically flagged and retired (Section 5).
2. **Against contamination**: every scored batch is freshly instantiated from seeded template generators; a small frozen "anchor" stream measures the memorization gap directly, and that gap is a published, score-relevant quantity (Section 4).
3. **Against format confounds**: a lenient, ordered extraction cascade with truncation-aware grading separates parsing from ability; format compliance is measured in its own domain, never silently everywhere (Section 8).
4. **Against scalar blending**: quality, cost, and latency are never merged; the efficiency frontier is published as a Pareto set, every score carries a 95% confidence interval, and per-domain scores are always available for user-side re-weighting (Section 9).
5. **Against opacity**: every displayed number traces to an immutable run record with stored raw responses; every model has a public page exposing every graded answer and every judge verdict; the methodology is versioned with a public changelog (Section 10).

2 Design Principles

Seven principles govern every design decision in the platform.

P1 — Measure ability, not accuracy. Accuracy conflates the difficulty of the item set with the capability of the model. Latent-ability estimation under an item response model separates the two and yields principled uncertainty (Section 5).

P2 — Generate, never fix. A test item that exists before evaluation time can be memorized. Items are therefore functions (seeded generators), not data. The template *code* is public — transparency lives at the level of the generating process — while instantiated items and answer keys remain server-side.

P3 — Grade mechanically wherever possible. Every closed-ended domain is graded by deterministic computation: normalized string equality, numeric tolerance, canonical-JSON equality of tool-call sequences, exact multi-line output matching, or constraint-checker stacks. LLM judges are confined to genuinely open-ended domains, and there they are treated as noisy instruments to be bias-controlled and audited, not as oracles (Section 6).

P4 — Target the information optimum. Item information peaks where solve probability is near one half. Generators expose difficulty knobs (composition depth, distractor density, counterfactual rules, context load) whose settings are chosen so strong models operate near that optimum.

P5 — Report uncertainty or report nothing. Every published score carries a 95% interval derived from the Fisher information of the underlying fit. Interfaces are required to render the interval wherever the score appears.

P6 — Never blend incommensurables. Quality sub-metrics with a shared construct (ability, stability, honesty, memorization-resistance) may be combined under published weights; cost and latency may not. The efficiency frontier is a set, not a number.

P7 — History is immutable; method is versioned. Runs are append-only. Corrections happen by new runs under a bumped semantic version with a changelog entry, never by editing the past.

3 System Overview

3.1 Architecture

LLMindex is a monorepo comprising a public web application and API; an evaluation worker executing batches, duels, and refits; a psychometrics engine (2PL and Bradley–Terry fitting with uncertainty quantification); a pure scoring library holding domain definitions, weights, and the index version as the single source of truth; and an item bank of template generators with their embedded simulators.

3.2 Unified model access

All evaluated models are reached through a single OpenAI-compatible aggregation layer (OpenRouter), which provides a uniform chat interface, per-token pricing metadata synchronized daily, and multimodal input for vision-capable models. Every call is issued with explicit generation parameters — temperature 0 for scored samples, the model's default temperature for consistency samples — and a completion budget of up to 16,384 tokens, capped per model by its context window so that reasoning-tuned models are never silently truncated into failure. Transient errors (HTTP 429 and 5xx) are retried up to five times with exponential backoff; a batch in which more than 2% of calls ultimately fail is flagged *degraded* and excluded from every fit until re-run.

3.3 Audit trail

Each model call persists an audit row containing the exact request parameters, the raw provider response, extracted answer, grading verdict, self-reported confidence, token usage, measured wall-clock latency, and measured cost. A ranking whose raw responses are not stored is, by the platform's own rules, invalid. This audit surface is public: every model's page exposes every graded answer of the current index version, and every judged duel with its verdict, judge identity, and position-swap flag.

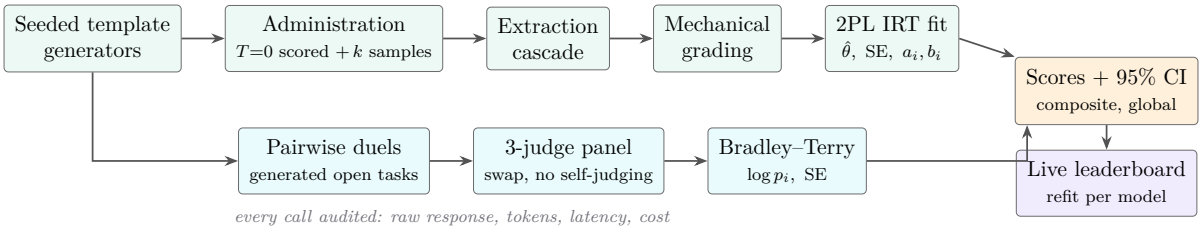

Figure 1: The LLMindex evaluation pipeline. Closed-ended domains flow through mechanical grading into the 2PL layer; open-ended domains flow through bias-controlled judged duels into the Bradley–Terry layer; both meet in a uniform score construction with published uncertainty.

Domain	Item family (generated)	Grading
code	program-trace prediction incl. nested control flow	numeric
math	deep chains, linear systems, counterfactual base-k, chained subproblems	numeric/exact
reasoning	7-entity order deduction, positional puzzles, decoy entities	exact
agentic	simulated tool-calling under policy; context-load ledgers	canonical JSON
terminal	simulated POSIX subset: pipelines, file trees, exit-code chains	exact lines
knowledge	free-response curated facts (no options, no guessing floor)	exact
multilingual	number words 0–999, French/Spanish, both directions	exact/numeric
instruction_following	repetition/acronym tasks; 5-constraint stacks	exact/checkers
vision_ocr	generated cluttered scenes, rasterized; table reading	exact/numeric
writing	constrained creative briefs	duels + BT
safety_refusal_quality	delicate gray-zone assistance scenarios	duels + BT
svg_design	reproduce real logos as raw SVG from memory	duels + BT

Table 1: The twelve scored domains of index version 0.2.0. The upper block is graded mechanically and scored by 2PL IRT; the lower block is ranked by cross-provider judged duels aggregated with Bradley–Terry.

3.4 Evaluation pipeline

Figure 1 shows the end-to-end flow from seeded generation to the live leaderboard.

3.5 Evaluation domains

Version 0.2.0 scores twelve domains (Table 1): nine closed-ended domains graded mechanically under the IRT layer, and three open-ended domains ranked by judged duels under the Bradley–Terry layer.

Two domain families deserve elaboration because they are fully home-made simulations. The **agentic** family presents a mock tool catalog salted with distractor tools, a world state, and deterministic business rules (escalation policies, overdraft pre-funding, dependency-ordered deployment); a built-in simulator computes the unique correct call sequence, and grading is exact canonical-JSON equality — a design informed by the state-comparison graders of τ-bench [29] and BFCL [19] but requiring no live infrastructure and admitting no reward hacking. A dedicated *context-load* template buries the handful of policy-relevant records inside a generated ledger of 120–300 near-miss decoys, making prompt length itself the difficulty knob.

The **terminal** family never executes a shell: a closed, deliberately unambiguous POSIX subset (fixed-string `grep`, `cut`, byte-order C-locale `sort`, integer-only `awk` aggregation, `head`/`tail`) is simulated in TypeScript, following the output-prediction paradigm of CRUXEval [9]; constructs with locale-dependent or implementation-divergent semantics are excluded by construction [8], so every answer is unambiguous, and the container-escape and answer-key-reading exploits documented against agent-executed terminal benchmarks [24] are impossible by design.

4 Item Generation and Contamination Resistance

4.1 Seeded template generators

An item template is a pure function $g_T(\sigma) \mapsto (\text{prompt}, \text{key}, \text{grading mode})$ from a seed string σ to an instantiated item. Seeds are expanded through a deterministic PRNG (a 32-bit mixing generator keyed by an FNV-1a hash of the seed), so any batch is exactly reproducible from its recorded seed, yet unpredictable before the run. Templates randomize surface values, entity names, paraphrase frames, and structural parameters; the space of instantiations per template ranges from 10^4 to beyond 10^{12}.

4.2 Difficulty engineering

Because generation is programmatic, difficulty is a controllable parameter rather than an accident of curation. The generators implement difficulty mechanisms whose discriminative power is documented in the literature: multi-step dependent computation in which errors compound multiplicatively [5]; chained sub-problems whose intermediate answers feed forward, converting small per-step differences into large end-to-end separations [31]; provably inert distractor clauses whose quantities never enter the computation [17, 23]; counterfactual rule perturbations such as base-k arithmetic that separate rule-following computation from memorized decimal facts [28]; large constraint-satisfaction and deduction spaces [15]; and stacked, individually verifiable output constraints [32]. Free-response formats are preferred over multiple choice wherever an unambiguous key exists, eliminating the guessing floor that both inflates scores and depresses item information.

4.3 Anchors and the measured memorization gap

A fixed fraction of each batch — at most 20%, enforced in code — derives from a *frozen* seed stream that is byte-identical in every run, forever. These anchor items serve longitudinal comparability and, more importantly, turn contamination from a suspicion into a measurement:

$$\Delta_{\text{contam}} = \text{acc}(\text{anchor items}) - \text{acc}(\text{fresh items}), \qquad R = \text{clamp}_{[0,1]}\left(1 - \frac{\Delta_{\text{contam}}}{0.2}\right), \quad (1)$$

where R is the contamination-resistance sub-metric entering the domain composite (Section 9). A model that performs 20 accuracy points better on the fixed stream than on structurally identical fresh instantiations receives zero resistance credit. Because anchors are never published and fresh items never repeat, the only path to a high score is capability on unseen instances.

5 Psychometric Scoring: the 2PL Layer

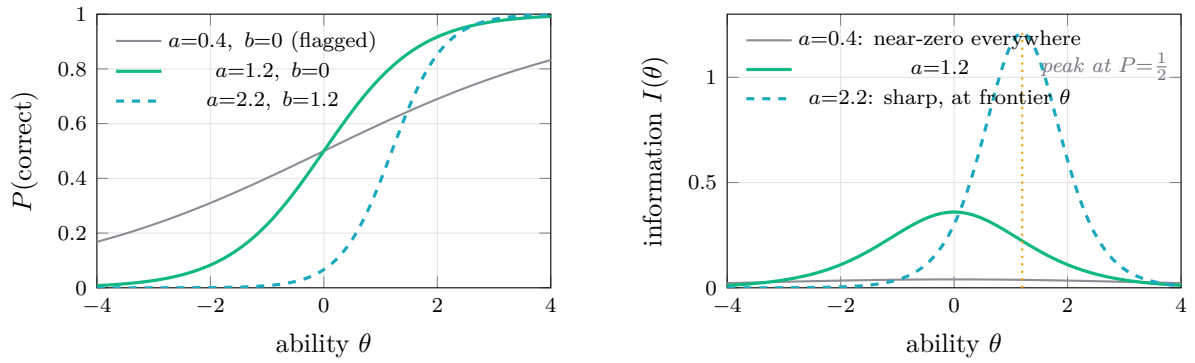

Figure 2: Left: 2PL item characteristic curves. Right: the corresponding item information $I(\theta) = a_i^2 P(1-P)$. A flat item (a=0.4) contributes almost nothing anywhere and is auto-flagged; a discriminative item placed at frontier ability (a=2.2, b=1.2) measures precisely where saturated benchmarks are blind.

5.1 Model

Let $x_{mi} \in \{0, 1\}$ indicate whether model m answered item i correctly. LLMindex adopts the two-parameter logistic model [1, 7]:

$$P(x_{mi} = 1 \mid \theta_m, a_i, b_i) \;=\; \sigma\big(a_i(\theta_m - b_i)\big) \;=\; \frac{1}{1 + e^{-a_i(\theta_m - b_i)}}, \tag{2}$$

where θ_m is the latent ability of model m, b_i the item difficulty (in logits), and $a_i > 0$ the item discrimination — the steepness with which the item separates abilities around b_i. Figure 2 illustrates both the item characteristic curves and the induced information functions.

5.2 Estimation

Parameters are estimated per domain by maximum a posteriori over the observed (possibly incomplete) response matrix:

$$\mathcal{L} = \sum_{(m,i)\,\in\,\text{obs}} \Big[x_{mi} \log P_{mi} + (1 - x_{mi}) \log(1 - P_{mi})\Big] - \sum_m \frac{\theta_m^2}{2\sigma_\theta^2} - \sum_i \frac{b_i^2}{2\sigma_b^2} - \sum_i \frac{(\log a_i)^2}{2\sigma_{\log a}^2}, \tag{3}$$

with priors $\theta \sim \mathcal{N}(0, 1)$, $b \sim \mathcal{N}(0, 1.5^2)$, and $\log a \sim \mathcal{N}(0, 0.5^2)$; the log-normal prior on a enforces positivity and regularizes the small-sample regime. Optimization uses Adam on the penalized likelihood with warm starts from row- and column-accuracy logits; abilities are re-centered to mean zero after each step (the shift being absorbed into b), with scale pinned by the priors. Convergence is declared at a relative objective change below 10^{-6}, capped at 500 iterations, and convergence diagnostics are stored with every fit.

5.3 Uncertainty

The standard error of each ability follows from the Fisher information of the 2PL likelihood at the estimate, plus the prior precision:

$$\mathrm{SE}(\theta_m) = \Big(\textstyle\sum_{i \in \mathrm{obs}(m)} a_i^2\, P_{mi}(1 - P_{mi}) \;+\; \sigma_\theta^{-2}\Big)^{-1/2}. \tag{4}$$

Equation (4) is also the design compass of the platform: the summand $a_i^2 P(1-P)$ is the *item information function*, maximal at $P = \frac{1}{2}$ and vanishing at the extremes. Generating items

whose difficulty places strong models near $P \approx 0.5$ is not sadism but statistics — it is where measurement is most efficient.

5.4 Item hygiene

After every refit, items with fitted discrimination $a_i < 0.3$ or difficulty $|b_i| > 3$ logits are automatically flagged for retirement review; flagged template-parameter regions are examined before subsequent runs. Dead items are thereby prevented from silently diluting the index — the failure mode that afflicts every static benchmark as models improve.

6 Pairwise Duels and the Bradley–Terry Layer

Open-ended capabilities — constrained creative writing, the quality of refusal behavior in delicate gray-zone situations, and the reproduction of real-world logos as raw SVG code — admit no mechanical key. For these domains LLMindex uses pairwise comparison, the measurement design with the longest pedigree in preference scaling [2, 6] and the design underlying Chatbot Arena [4], but with LLM judges under explicit bias controls in place of crowdsourced votes.

6.1 Judge protocol

Each duel presents the same generated task to two models; their responses are then rated by a panel of three judge models drawn from at least two different providers. The protocol enforces: (i) *no self-judging* — a judge that is a duel participant is excluded from that duel, and duels with fewer than two eligible judges are discarded; (ii) *position swapping* — judges alternate which response appears first, the swap is recorded on every verdict row, and verdicts are un-swapped before aggregation, rendering position bias both mitigated and measurable; (iii) *conservative parsing* — judges answer with a single verdict token at temperature 0, unparseable verdicts count as ties, and empty or degenerate responses can never win; (iv) *anti-verbosity instruction* — judge prompts explicitly forbid rewarding length or style over substance. Panel agreement rates are computed per run and published in the run's immutable diagnostics.

6.2 Aggregation

Verdicts feed a Bradley–Terry model: model i beats model j with probability $p_i/(p_i + p_j)$, ties contributing half a win to each side. Strengths are fitted with the minorization–maximization iteration [12]

$$p_i \leftarrow W_i \Big/ \sum_{j \neq i} \frac{N_{ij}}{p_i + p_j}, \tag{5}$$

where W_i is the (fractional) win count of model i and N_{ij} the number of comparisons between i and j; a small damping term (0.1 phantom wins per pair) keeps strengths finite for sparsely compared models. Uncertainty follows from the Fisher information of the log-strength parametrization,

$$\mathcal{I}_{ii} = \sum_{j \neq i} N_{ij}\, p_{ij}(1 - p_{ij}), \qquad p_{ij} = \frac{p_i}{p_i + p_j}, \qquad \mathrm{SE}(\log p_i) = \mathcal{I}_{ii}^{-1/2}. \tag{6}$$

Log-strengths are standardized to zero mean and unit variance, placing duel domains on the same θ-scale as the IRT domains before rescaling; downstream score construction is thereby uniform across both layers.

7 Robustness Metrics

Three additional measurements enter each domain's composite, each capturing a failure mode invisible to accuracy.

7.1 Consistency

Every scored item is also administered k additional times at the model's default sampling temperature. With A_{mi} the multiset of extracted answers of model m on item i,

$$\text{consistency}(m) = \frac{1}{|I_m|} \sum_{i \in I_m} \frac{\max_v \#\{a \in A_{mi} : a = v\}}{|A_{mi}|}, \tag{7}$$

the mean share of samples agreeing with the modal answer. Large accuracy variance across re-instantiations of identical templates is a documented phenomenon [17]; a model that flips answers under resampling is less trustworthy than its accuracy suggests, and this metric prices that in.

7.2 Calibration

Every graded item requires the model to append a confidence self-report on a 0–100 scale — verbalized confidence being better calibrated than token likelihoods for RLHF-trained models [25]. With confidences binned into ten equal-width bins $B_1, \ldots, B_{10}$,

$$\text{ECE} = \sum_{b=1}^{10} \frac{|B_b|}{N} \Big| \text{acc}(B_b) - \overline{\text{conf}}(B_b) \Big|, \qquad \text{calibration} = 1 - \text{ECE}, \tag{8}$$

following Guo et al. [10]; the Brier score [3] is computed alongside as a diagnostic. Missing confidences are treated as missing, not imputed at 50, so calibration is never polluted by non-compliance — which is measured elsewhere.

7.3 Contamination resistance

As defined in Equation (1), the anchor-versus-fresh accuracy gap maps to a resistance score with a hard floor: a gap of 20 accuracy points or more earns zero credit. Unlike post-hoc contamination detection, this quantity is measured *inside* the evaluation, per model, per domain, on every run.

8 Answer Extraction and Grading

Grading robustness is treated as a first-class methodological concern. Extraction proceeds through an ordered, lenient cascade: the last markdown-stripped line matching an answer tag (`ANSWER:`, `FINAL ANSWER:`, with any bold/backtick wrapping absorbed); failing that, the last LaTeX `\boxed{}` expression; for structured answers (tool-call sequences, terminal output), the last fenced code block with a balanced-bracket JSON scan and repair pass. Numeric grading normalizes thousands separators, currency symbols, unit suffixes, and Unicode minus signs, and compares under relative tolerance 10^{-6}; string grading normalizes case, punctuation, and hyphen/space orthographic variants; tool-call grading canonicalizes JSON (sorted keys, whitespace-free) before exact comparison; constraint-stack items are validated by deterministic checker functions, so any of the astronomically many conforming outputs grades correct.

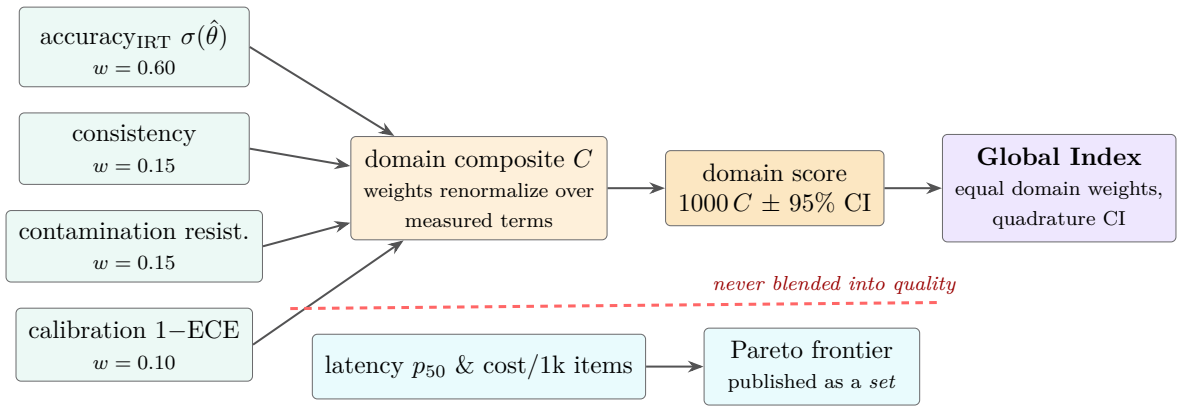

Figure 3: Score construction. Four commensurable quality sub-metrics blend under published weights into the domain composite; economics stays below the red line, summarized only as a Pareto set.

Two policies deserve emphasis. First, *truncation is not failure*: a completion cut off at the token budget with no extractable answer is recorded as unscored, never as incorrect, because grading a severed chain of thought measures the budget rather than the model. Second, *format compliance is measured where it belongs*: the instruction-following domain explicitly tests constraint adherence, so leniency elsewhere does not erase the signal — it relocates it to the construct actually being claimed.

9 Score Structure and Interpretation

9.1 From sub-metrics to domain scores

Each closed-ended domain yields an ability estimate $\hat{\theta}$ with standard error, plus the three robustness metrics. The domain composite in $[0, 1]$ is

$$C \;=\; \frac{w_{\text{acc}}\,\sigma(\hat{\theta}) + w_{\text{con}}\cdot\text{consistency} + w_{\text{cal}}\cdot\text{calibration} + w_{\text{res}}\cdot R}{\sum_{\text{measured}} w}, \tag{9}$$

with weights $(w_{\text{acc}}, w_{\text{con}}, w_{\text{res}}, w_{\text{cal}}) = (0.60,\ 0.15,\ 0.15,\ 0.10)$ and renormalization over whichever sub-metrics were actually measured, so a missing measurement neither rewards nor punishes. The weight rationale is explicit: accuracy-as-ability is the primary construct and dominates; consistency and contamination resistance are robustness corrections grounded in documented failure modes; calibration, the noisiest of the four at current sample sizes, receives the smallest weight. The displayed domain score is $1000\,C$, with a 95% interval propagated from $\text{SE}(\hat{\theta})$ by the delta method through the accuracy term:

$$\text{half-width} \;=\; 1000 \cdot \frac{w_{\text{acc}}}{\sum w} \cdot \sigma(\hat{\theta})(1-\sigma(\hat{\theta})) \cdot 1.96 \cdot \text{SE}(\hat{\theta}). \tag{10}$$

Duel domains enter the same pipeline with the standardized Bradley–Terry log-strength in place of $\hat{\theta}$. Figure 3 summarizes the construction and the deliberate wall between quality and economics.

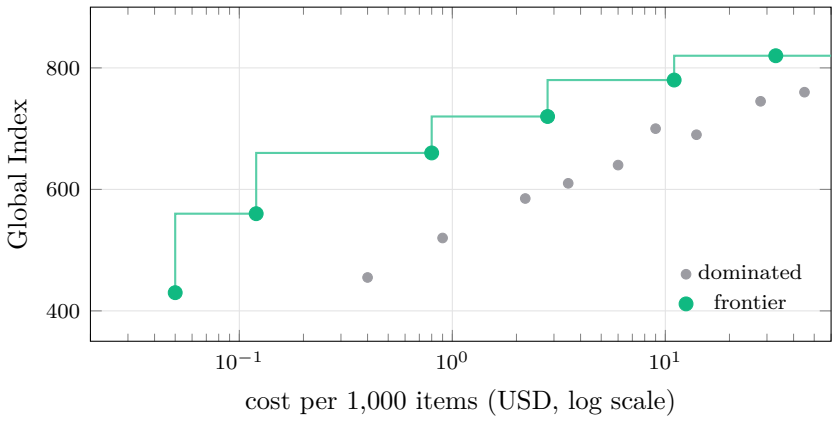

Figure 4: Illustrative efficiency frontier. Frontier models (green) are undominated on quality and cost simultaneously; every other model is strictly worse on both axes than some frontier point. The frontier is published as a set — deliberately never collapsed into a single "value" number.

9.2 The Global Index

The Global Index is the weighted mean of a model's domain composites over the domains it was actually evaluated on, rescaled to $[0, 1000]$:

$$G \;=\; 1000 \cdot \frac{\sum_{d \in D_m} w_d \, C_d}{\sum_{d \in D_m} w_d}, \qquad \text{half-width}(G) = \Big(\sum_d \big(w'_d \, h_d\big)^2 \Big)^{1/2}, \tag{11}$$

where h_d are domain half-widths, w'_d normalized weights, and independence across domain fits justifies combination in quadrature. Domain weights are *equal* — deliberately. Absent a task-utility function, any unequal weighting is an editorial value judgment; the maximum-entropy prior is the only non-arbitrary default, and because per-domain scores are always published, any reader with a genuine utility function can re-weight in seconds.

9.3 The efficiency frontier

For each model the platform publishes measured median latency and measured cost per thousand items (from metered token usage and live per-token pricing). The quality–cost plane is summarized by its Pareto frontier: the set of models not dominated on both axes (Figure 4). No blended "value score" exists anywhere on the platform, by principle P6.

9.4 How to read the numbers

- **Overlapping intervals mean no established order.** A 15-point gap under 60-point intervals is noise, and the interface will show exactly that.
- **Scores are version-scoped.** A score under index version 0.2.0 is not comparable to one under 0.1.0; the version is printed beside every ranking, and fits never mix versions.
- **Domain profiles beat the Global Index for decisions.** The Global Index summarizes; procurement should read the domain radar and the sub-metrics (a high-accuracy, poorly calibrated model is a specific risk profile).
- **Duel scores are comparative.** Bradley–Terry strengths locate models relative to the evaluated pool, not on an absolute craft scale.

- **Anchor-gap tells you about training data.** A large published Δ_{contam} for a model is direct evidence that its static-benchmark results elsewhere should be discounted.

10 Governance, Versioning, and Transparency

Semantic versioning of method. Domains, weights, hyperparameters, and grading rules live in a single source-of-truth module whose version follows semver. Any change bumps the version and appends a public changelog entry; the machine-readable configuration is served by the public API.

Immutable runs. Every evaluation batch, duel batch, and index fit is a database row carrying its item-set hash, model set, index version, status, and fit diagnostics. Historical runs are never edited. Corrections — including successful score disputes — are executed as *new* runs and acknowledged in the changelog.

Total response transparency. Each model's public page lists every graded answer of the current version (verdict, confidence, latency, cost, and the full prompt for non-anchor items) and every judged duel (judge identity, position-swap flag, verdict). Two things are withheld, on stated principle: answer keys, and anchor-item prompts — publishing either would destroy the measurement.

Independence. The platform evaluates models from all providers under identical protocols, takes no compensation from evaluated vendors, and publishes the judge configuration. Disputes are received at contact@spboucher.ai and adjudicated on the stored evidence.

11 Known Limitations

Intellectual honesty requires stating what the platform does *not* establish.

Small-examinee-population IRT. Item-parameter recovery in IRT degrades when the examinee population is small; with on the order of 10^2 models, item parameters are regularized by priors rather than pinned by data. The roster deliberately spans very weak to frontier models to anchor the scale, and score intervals are published precisely so users do not over-read; nevertheless, item-level parameters should be treated as noisier than the ability estimates built on them.

Judge subjectivity survives bias control. Position swapping, cross-provider panels, self-judging exclusion, and agreement reporting mitigate — but cannot eliminate — shared aesthetic preferences among LLM judges. Duel-domain scores should be read comparatively and alongside the published agreement rates.

Provider routing variance. Models are reached through an aggregation layer whose upstream routing can affect latency and, occasionally, behavior. Latency is reported as measured, not normalized to reference hardware.

Construct coverage. Generated items measure deep, narrow, mechanically-gradeable slices of each capability. This is a deliberate trade — breadth is the enemy of grading validity — but it means the index does not measure, e.g., long-horizon interactive tool use with error recovery (single-shot planned sequences are graded today), open-domain factuality at web scale, or multi-session memory. The template inventory is public; readers can judge coverage directly.

Anchor stream aging. Anchors resist contamination only until models train on traffic that includes them. Anchor rotation with overlap (retiring streams gradually while preserving longitudinal linkage) is on the roadmap.

Consistency confound. The consistency metric samples at each model's default temperature, which providers set differently; part of the observed stability difference is configuration rather than capability. It is reported as measured, with this caveat documented.

12 Comparison with Existing Efforts

Effort	Measurement	Contamination stance	Relation to LLMindex
MMLU / static suites [11, 26]	raw accuracy, fixed items	fixed public sets; saturated at the frontier	LLMindex replaces raw accuracy with 2PL ability and fixed items with generators
HELM [14]	multi-metric, fixed scenarios	fixed sets	shares multi-metric philosophy; LLMindex adds IRT, generation, and per-score CIs
Chatbot Arena [4]	human pairwise votes, BT/Elo	organic prompts; style confounds documented	same aggregation family for duels; LLMindex judges are bias-controlled panels on generated tasks, and duels are one layer, not the whole index
LiveBench [27]	accuracy on refreshed items	periodic refresh	LLMindex regenerates *every batch* and measures the memorization gap explicitly
tinyBenchmarks / metabench [13, 21]	IRT on existing benchmark items	inherits source-set exposure	validates the IRT layer; LLMindex applies it to unexposed generated items
BFCL / τ-bench [19, 29]	AST/state-graded tool use	fixed task sets	inspiration for simulator-graded agentic items, re-implemented as seeded generators
Terminal-Bench [24]	containerized tasks, test scripts	fixed tasks; grader-exploit incidents documented	LLMindex predicts simulated-shell outcomes: no container, no readable answer key, no exploit surface
GAIA / HLE [16, 20]	exact-match hard questions	held-out but fixed	shares frontier-difficulty targeting; LLMindex achieves it with regenerable items

Table 2: Positioning relative to representative evaluation efforts.

Table 2 summarizes. The synthesis is the contribution: psychometric scoring *on* contamination-proof generated items, duels *as one bias-controlled layer* rather than the whole story, robustness and honesty metrics *inside* the score, efficiency *outside* it, and an audit trail underneath everything.

13 Roadmap

1. **Adaptive administration.** Selecting the next item family and difficulty bin to maximize expected Fisher information at the model's running $\hat{\theta}$, manufacturing items at requested difficulty — the capability fixed benchmarks cannot have.
2. **Multi-turn agentic episodes.** Interactive tool loops against the deterministic simulators with injected typed faults, grading final state, required reads, and recovery under a strict failed-call budget.
3. **Anchor rotation with overlap.** Gradual retirement of anchor streams with linking items to preserve longitudinal comparability while bounding anchor exposure.
4. **Knob-to-difficulty calibration.** Learning the mapping from generator parameters to fitted b per template family, enabling item manufacture at prescribed difficulty.
5. **Judge audit corpus.** A fixed, hand-adjudicated duel set for measuring judge-panel drift across judge model upgrades.
6. **Public data releases.** Periodic dumps of expired (non-anchor) items with full response matrices, enabling third-party psychometric reanalysis.
7. **Third-party methodology audits** and a standing correction bounty on grading errors.

14 Conclusion

Benchmarking is measurement, and measurement has a discipline. LLMindex applies that discipline end to end: items that cannot be memorized because they do not exist until asked; ability estimated where information lives rather than where accuracy saturates; judges treated as instruments with quantified bias, not oracles; honesty about uncertainty in every number printed; a hard wall between quality and economics; and an audit trail that makes every claim checkable and every error correctable in public. The platform is live, its methodology is versioned, and its mistakes — when found — will be fixed in the open.

Correspondence: Simon-Pierre Boucher, contact@spboucher.ai.

References

[1] Allan Birnbaum. Some latent trait models and their use in inferring an examinee's ability. In Frederic M. Lord and Melvin R. Novick, editors, *Statistical Theories of Mental Test Scores*. Addison-Wesley, 1968.

[2] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[3] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[4] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, et al. Chatbot arena: An open platform for evaluating LLMs by human preference. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

[5] Nouha Dziri, Ximing Lu, Melanie Sclar, et al. Faith and fate: Limits of transformers on compositionality. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[6] Arpad E. Elo. The rating of chessplayers, past and present. *Arco Publishing*, 1978.

[7] Susan E. Embretson and Steven P. Reise. *Item Response Theory for Psychologists*. Lawrence Erlbaum Associates, 2000.

[8] Michael Greenberg and Austin J. Blatt. Executable formal semantics for the POSIX shell. In *Proceedings of the ACM on Programming Languages (POPL)*, 2020.

[9] Alex Gu, Baptiste Roziere, Hugh Leather, Armando Solar-Lezama, Gabriel Synnaeve, and Sida I. Wang. CRUXEval: A benchmark for code reasoning, understanding and execution. *arXiv preprint arXiv:2401.03065*, 2024.

[10] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

[11] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*, 2021.

[12] David R. Hunter. MM algorithms for generalized Bradley-Terry models. *The Annals of Statistics*, 32(1):384–406, 2004.

[13] Alex Kipnis, Konstantinos Voudouris, Luca M. Schulze Buschoff, and Eric Schulz. metabench: A sparse benchmark of reasoning and knowledge in large language models. In *International Conference on Learning Representations (ICLR)*, 2025.

[14] Percy Liang, Rishi Bommasani, Tony Lee, et al. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023.

[15] Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, et al. ZebraLogic: On the scaling limits of LLMs for logical reasoning. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

[16] Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: A benchmark for general AI assistants. *arXiv preprint arXiv:2311.12983*, 2023.

[17] Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. GSM-Symbolic: Understanding the limitations of mathematical reasoning in large language models. *arXiv preprint arXiv:2410.05229*, 2024.

[18] Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B. Hashimoto. Proving test set contamination in black box language models. *International Conference on Learning Representations (ICLR)*, 2024.

[19] Shishir G. Patil, Huanzhi Mao, Charlie Cheng-Jie Ji, et al. The Berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

[20] Long Phan, Alice Gatti, Ziwen Han, et al. Humanity's last exam. *Nature*, 2025.

[21] Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: Evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

[22] Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design. In *International Conference on Learning Representations (ICLR)*, 2024.

[23] Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023.

[24] The Terminal-Bench Team. Terminal-bench: A benchmark for AI agents in terminal environments. *https://www.tbench.ai*, 2025.

[25] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of EMNLP*, 2023.

[26] Yubo Wang, Xueguang Ma, Ge Zhang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[27] Colin White, Samuel Dooley, Manley Roberts, et al. LiveBench: A challenging, contamination-free LLM benchmark. *arXiv preprint arXiv:2406.19314*, 2024.

[28] Zhaofeng Wu, Linlu Qiu, Alexis Ross, et al. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of NAACL*, 2024.

[29] Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

[30] Qingchen Yu, Zifan Zheng, Shichao Song, et al. xFinder: Robust and pinpoint answer extraction for large language models. *arXiv preprint arXiv:2405.11874*, 2024.

[31] Sitong Yu et al. R-Horizon: How far can your large reasoning model really go in breadth and depth? *arXiv preprint arXiv:2510.08189*, 2025.

[32] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, et al. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.