UNDERSTANDING AI · THE WRITTEN EDITION · 2026-09-02
Day 1 · typeset 2026-09-02 09:41Z

In July a benchmark score nearly tripled without anything inside the model changing. The interesting question is not how the software was improved. It is why a number that everyone treats as a property of a model turned out to be a property of the software around it — and why computing worked that out, and wrote it into its rules, nearly forty years ago.

1. A note about two settings

On 29 July 2026 two OpenAI engineers, Ilan Bigio and Ted Sanders, published a short note under the title *"How enabling two settings tripled our scores on the ARC-AGI-3 benchmark"*. Its subject was a set of small video games that a computer system has to learn by playing them, built by the ARC Prize Foundation. Under the runner ARC Prize supplies to every provider, the company's model scored 13.3 on the public demonstration set. Under a runner OpenAI rebuilt against its own interface, with two settings turned on, the same model scored 38.3 — and produced roughly a sixth as many output tokens per game.

The model was not retrained. The checkpoint and the reasoning tier were the same in both runs.

Figure 1. The July result, in four numbers

13.3
Score, ARC Prize's runner
public demonstration set, max reasoning effort

38.3
Score, OpenAI's rebuilt runner
same checkpoint, same reasoning tier, no retraining

2.88×
The ratio
OpenAI's own body text says "roughly 3x"

5.98×
Fewer output tokens per game
at max effort only; at low effort the rebuilt runner produced 43% more

OpenAI, How enabling two settings tripled our scores on the ARC-AGI-3 benchmark, 29 July 2026. The two ratios are computed from the ten datapoints embedded in that page's own chart: 0.383/0.133 = 2.879 and 2,900,997/485,485 = 5.975.

▼ **Table view**

Figure 1. The July result, in four numbers

Measure	Value
Score, ARC Prize's runner	13.3
Score, OpenAI's rebuilt runner	38.3
The ratio	2.88×
Fewer output tokens per game	5.98×

2. Two readings, both incomplete

The first reading is that the benchmark is meaningless. If a number nearly triples because somebody rewrote the software around a fixed model, the number was never measuring anything. That reading throws away a real measurement: something was held constant and something was varied, and the difference is informative about the thing that varied.

The second is the opposite — that the model was better than the benchmark said, and the test had been unfair to it. That reading is checkable, and it fails on the record. ARC Prize's leaderboard reports scores on a held-out set, and the 38.3 is on the public demonstration set. The two are different collections of environments, so the improvement cannot be read as a change in leaderboard standing.

Both readings share an assumption: that one thing is being tested and the number belongs to it. Almost no published figure about an interactive AI system is a measurement of one thing.

3. What is actually being measured

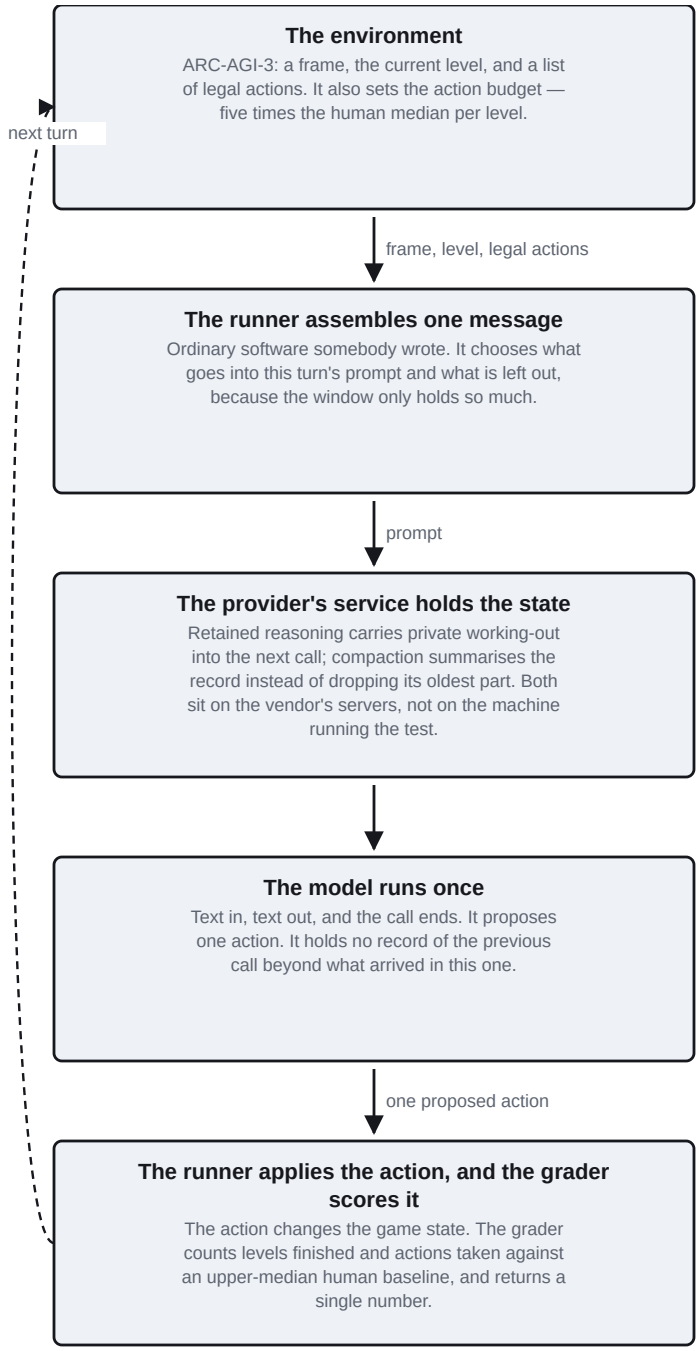
A trained model is something a program calls. Text goes in, text comes out, and the call ends. It retains what it learned during training, and it can use whatever is placed in front of it on that call. What it does not retain is the case: it holds no record of the previous call unless a piece of software puts that record back in front of it. It does not open files, run programs, or decide to continue.

Software that writes code for an hour therefore cannot be the model alone. Around it sits ordinary software — a loop, a store of state, a set of rules about what to send and what to withhold — which the field generally calls the **harness**. The harness chooses what the model sees, which actions it may take, what it remembers, and when the job ends. Those are decisions, and they are not made by the model.

Scaffolding is a narrower word for a part of the harness: the pieces installed because the model could not do something reliably alone — a planner, a summariser, a step that checks the work. Scaffolding is compensation, which is why it is the part that can later be removed.

The **system** is the whole declared arrangement: the model, the harness, the tools it can reach, the resource and action limits it runs under, and the grader that scores the outcome. A product is a system with a name, permissions and a company attached.

Figure 2. One turn, and who decides what



Schematic, drawn from OpenAI's note of 29 July 2026 and the ARC-AGI-3 technical report (arXiv 2603.24621v2). The middle stage is the one a two-object description has nowhere to put: ARC Prize's reply says its own runner manages conversation state client-side, and the change OpenAI made moved that state to the provider. The score is a property of the whole loop, not of the fourth box.

▼ **Table view**

Figure 2. One turn, and who decides what — stages

#	Stage	Note
1	The environment	ARC-AGI-3: a frame, the current level, and a list of legal actions. It also sets the action budget — five times the human median per level.

#	Stage	Note
2	The runner assembles one message	Ordinary software somebody wrote. It chooses what goes into this turn's prompt and what is left out, because the window only holds so much.
3	The provider's service holds the state	Retained reasoning carries private working-out into the next call; compaction summarises the record instead of dropping its oldest part. Both sit on the vendor's servers, not on the machine running the test.
4	The model runs once	Text in, text out, and the call ends. It proposes one action. It holds no record of the previous call beyond what arrived in this one.
5	The runner applies the action, and the grader scores it	The action changes the game state. The grader counts levels finished and actions taken against an upper-median human baseline, and returns a single number.

Figure 2. One turn, and who decides what — connections

From	To	Label
The environment	The runner assembles one message	frame, level, legal actions
The runner assembles one message	The provider's service holds the state	prompt
The provider's service holds the state	The model runs once	
The model runs once	The runner applies the action, and the grader scores it	one proposed action
The runner applies the action, and the grader scores it	The environment	next turn

The rule that follows is the one worth keeping. Where the line between model and system is drawn is a choice made for a question, not a fact discovered in nature. To compare models, hold the surrounding software as fixed as possible. To compare products, let it vary and disclose it. To attribute an improvement to a component, hold everything else constant and change one thing. And whichever line is drawn, the score belongs to everything inside it.

4. Computing settled this in 1988

None of that is new, and none of it originates in artificial intelligence.

In 1988 a group of workstation vendors founded the Standard Performance Evaluation Corporation, in its own account, "because they recognized the desperate need for realistic, standardized performance tests". The problem it faced was exactly the present one: the same processor produced different numbers under a different compiler, a different operating system or different memory, and vendors quoted whichever number suited them. The remedy was not to abandon measurement. It was to require that a result arrive with the conditions attached. Rule 1.2.2 of the current SPEC CPU run rules is headed *Conditions of Observation*:

> The report that certain performance has been observed is meaningful only if the conditions of observation are stated. SPEC therefore requires that a published result include a description of all performance-relevant conditions.

The same rules name the thing being measured the **System Under Test**, and require that a later tester be able to obtain the described components and reproduce the result within run-to-run variation. Four years later the Text REtrieval Conference, begun in 1992 under NIST and the US Department of Defense, fixed the same problem from the other end: rather than requiring each entrant to disclose its configuration, it held the task itself constant — participants ran their own systems on a shared collection of documents and questions and returned ranked results to a single central scorer. Disclosure and standardisation are different instruments, and both make a score belong to a declared arrangement rather than to a component.

Agent benchmarks have inherited the problem and have not yet inherited the paperwork. There is no configuration sheet beside an ARC-AGI-3 score, and the July dispute is what that absence looks like when it surfaces.

5. What changed, exactly

The headline says two settings. The record shows more than two things changed, and the difference matters for what may be concluded.

	ARC Prize's runner	OpenAI's rebuilt runner
Model checkpoint	GPT-5.6 Sol	**unchanged**
Reasoning effort	max	**unchanged**
Training	—	**none**
Interface	the completions-style API, state held client-side	rebuilt against OpenAI's Responses API, **state held by the provider**
Private reasoning between actions	discarded after every action	**retained**
Record when the window filled	oldest messages dropped	**summarised**
Truncation threshold	175,000 characters	175,000 **tokens**

OpenAI describes the last of those as immaterial — its note says the token limit "ends up being quite similar" to the character limit, because most of the text is action grids that its tokeniser splits about one-to-one. That is the party that ran the experiment assessing the size of its own confound, and no outside measurement of it exists.

The second row from the bottom is the one a two-object account cannot hold. ARC Prize's reply describes its own arrangement as managing conversation state client-side; what OpenAI changed moved that state onto the provider's servers. Memory did not appear inside the model. It moved from one piece of software to another, across a company boundary.

On the first two changes the note is precise, and the precision is worth preserving:

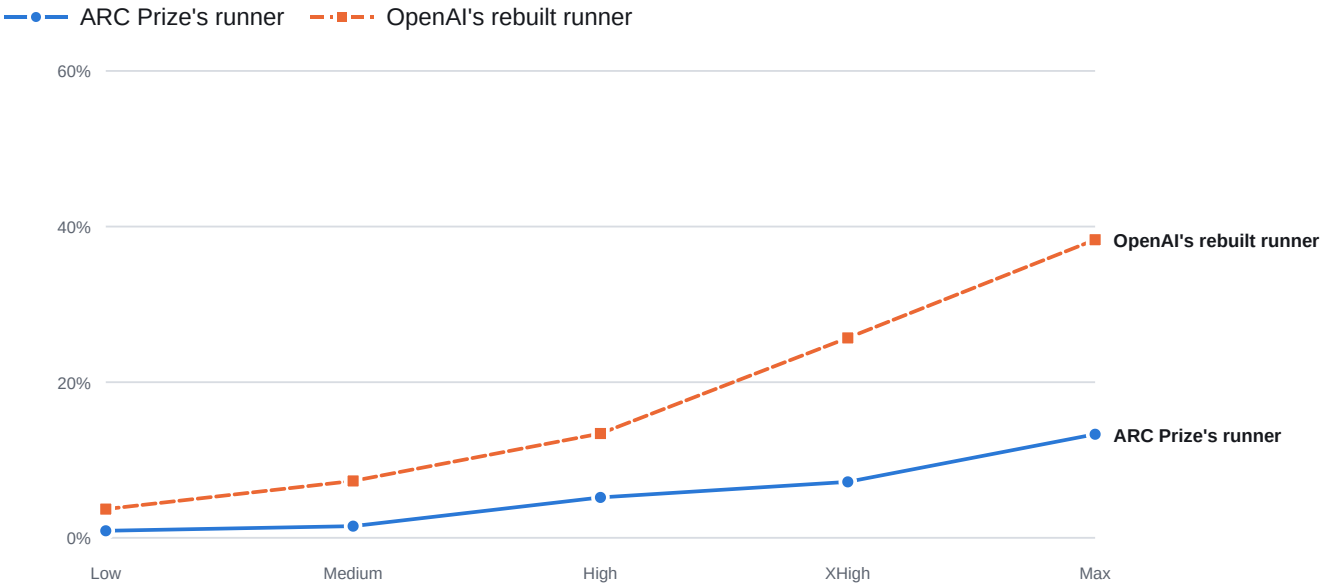
> First, we noticed that after each game action, all private reasoning was discarded. This meant that with each action, GPT-5.6 Sol was asked to figure out the game anew, unable to remember its past thinking. The model could still see a record of past moves and brief accompanying notes, but it could not see the plans, insights, or thoughts that led to them.
>
> Second, we saw that the harness used a rolling truncation window, causing older actions to become invisible as the history grew.

A diary of moves survived; the reasoning behind them did not, and the oldest entries fell off the end.

6. The whole curve, not the headline pair

The two numbers in the headline are one of five pairs. OpenAI ran both configurations at five reasoning-effort levels and plotted all ten points; the underlying values are embedded in the published chart and can be read off it directly.

Figure 3. Score at each reasoning effort, both runners



OpenAI, 29 July 2026; values extracted from the datapoint labels of that page's own chart on 2026-09-02 and archived. Score is Relative Human Action Efficiency on the 25-environment public demonstration set. The rebuilt runner leads at every effort level; the ratio ranges from 2.58× to 4.87×, so "roughly tripled" describes the whole sweep and not only the headline pair.

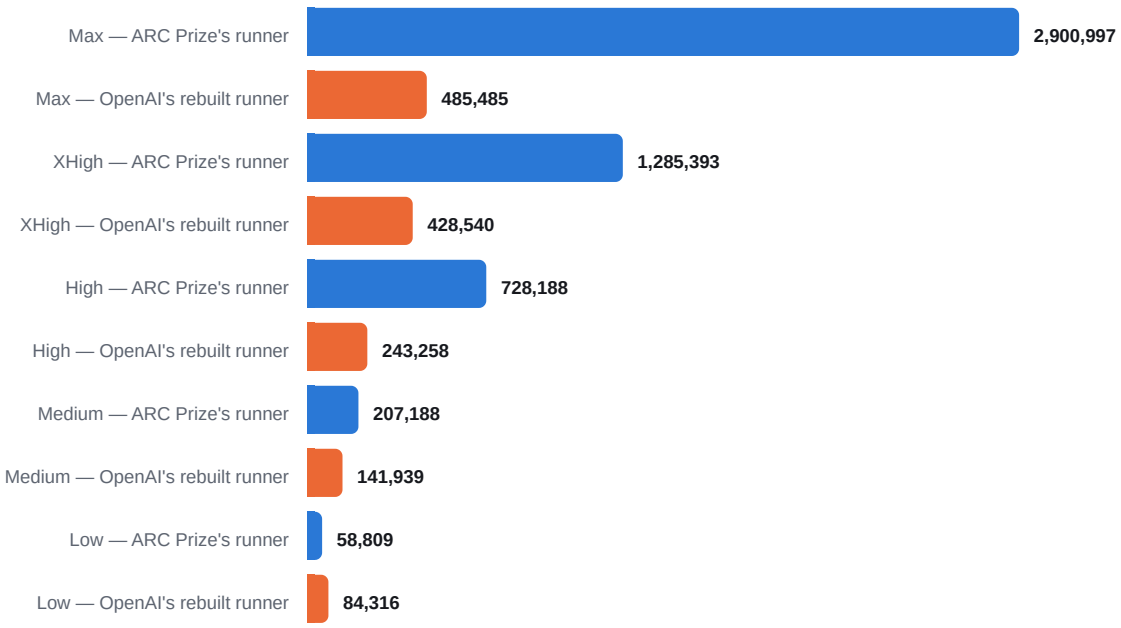
▼ **Table view**

Figure 3. Score at each reasoning effort, both runners

Reasoning effort	ARC Prize's runner	OpenAI's rebuilt runner
Low	0.9%	3.7%
Medium	1.5%	7.3%
High	5.2%	13.4%
XHigh	7.2%	25.7%
Max	13.3%	38.3%

The token claim is narrower than the score claim. Measured in output tokens per game, the rebuilt runner is cheaper at high, extra-high and max effort — and *more* expensive at low effort, where it produced about 43% more text than the runner it replaced.

Figure 4. Output tokens per game, by reasoning effort and runner



Same source and extraction as Figure 3. Ratios, old over new: 5.98× at max, 3.00× at extra-high, 2.99× at high, 1.46× at medium and 0.70× at low. The headline six-fold is the most favourable of the five. These are output tokens, not money: no cost figure for either run has been published.

▼ **Table view**

Figure 4. Output tokens per game, by reasoning effort and runner

Configuration	Output tokens per game
Max — ARC Prize's runner	2,900,997
Max — OpenAI's rebuilt runner	485,485
XHigh — ARC Prize's runner	1,285,393
XHigh — OpenAI's rebuilt runner	428,540
High — ARC Prize's runner	728,188
High — OpenAI's rebuilt runner	243,258
Medium — ARC Prize's runner	207,188
Medium — OpenAI's rebuilt runner	141,939
Low — ARC Prize's runner	58,809
Low — OpenAI's rebuilt runner	84,316

The most economical statement of the harness effect is one the note does not make. At *high* effort the rebuilt runner scored 13.4 on 243,258 output tokens per game. At *max* effort ARC Prize's runner scored 13.3 on 2,900,997. The same score, for about a twelfth of the output.

7. What the number is, and which numbers may be compared

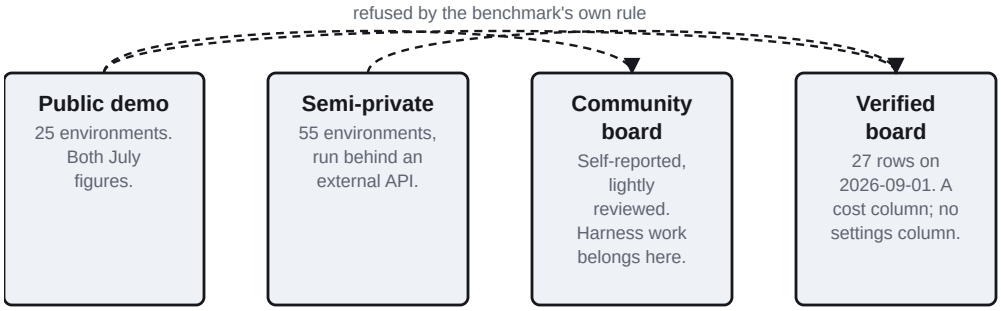
The score is not a percentage of puzzles solved. Relative Human Action Efficiency, defined in the ARC-AGI-3 technical report, takes for each level the ratio of a human baseline action count to the agent's action count, **squares it**, caps it at 1.15, weights later levels more heavily than earlier ones, and then caps the environment score by the fraction of levels actually completed. The baseline is the upper-median best human action count, not an average player. A score of 38.3 is a position on that constructed scale; it does not mean 38% of anything.

Which set the number came from matters as much as what the number means. ARC-AGI-3 has three: a public demonstration set of 25 environments, and semi-private and fully private sets of 55 each. The technical report is unambiguous about the first:

> Because it is impossible to ensure that system designers don't use the public environments as part of their work, and because the public set is materially easier than the private set, we will never report public set scores of any system on the official leaderboard.

That sentence is from the benchmark's authors, about the set both July figures were measured on. ARC Prize's current Verified Testing Policy says the opposite about relative difficulty — that for ARC-AGI-3 "the public demo is harder than the Semi-Private set" — a contradiction that has stood in its published material since at least mid-July and that neither document acknowledges. The measurements favour the technical report: the same model at max effort averages 13.33% on the public set and 7.78% on the semi-private one. The adjective is not load-bearing either way, because the prohibition on leaderboarding a public-set score does not depend on it.

Figure 5. Which score may travel to which board



ARC-AGI-3 technical report (arXiv 2603.24621v2) and arcprize.org, read 2026-09-02. The dashed arrow is the one the report forbids, and splicing a public-set figure against a semi-private one is the error still circulating in coverage of the July result.

▼ **Table view**

Figure 5. Which score may travel to which board — stages

#	Stage	Note
1	Public demo	25 environments. Both July figures.
2	Semi-private	55 environments, run behind an external API.
3	Community board	Self-reported, lightly reviewed. Harness work belongs here.
4	Verified board	27 rows on 2026-09-01. A cost column; no settings column.

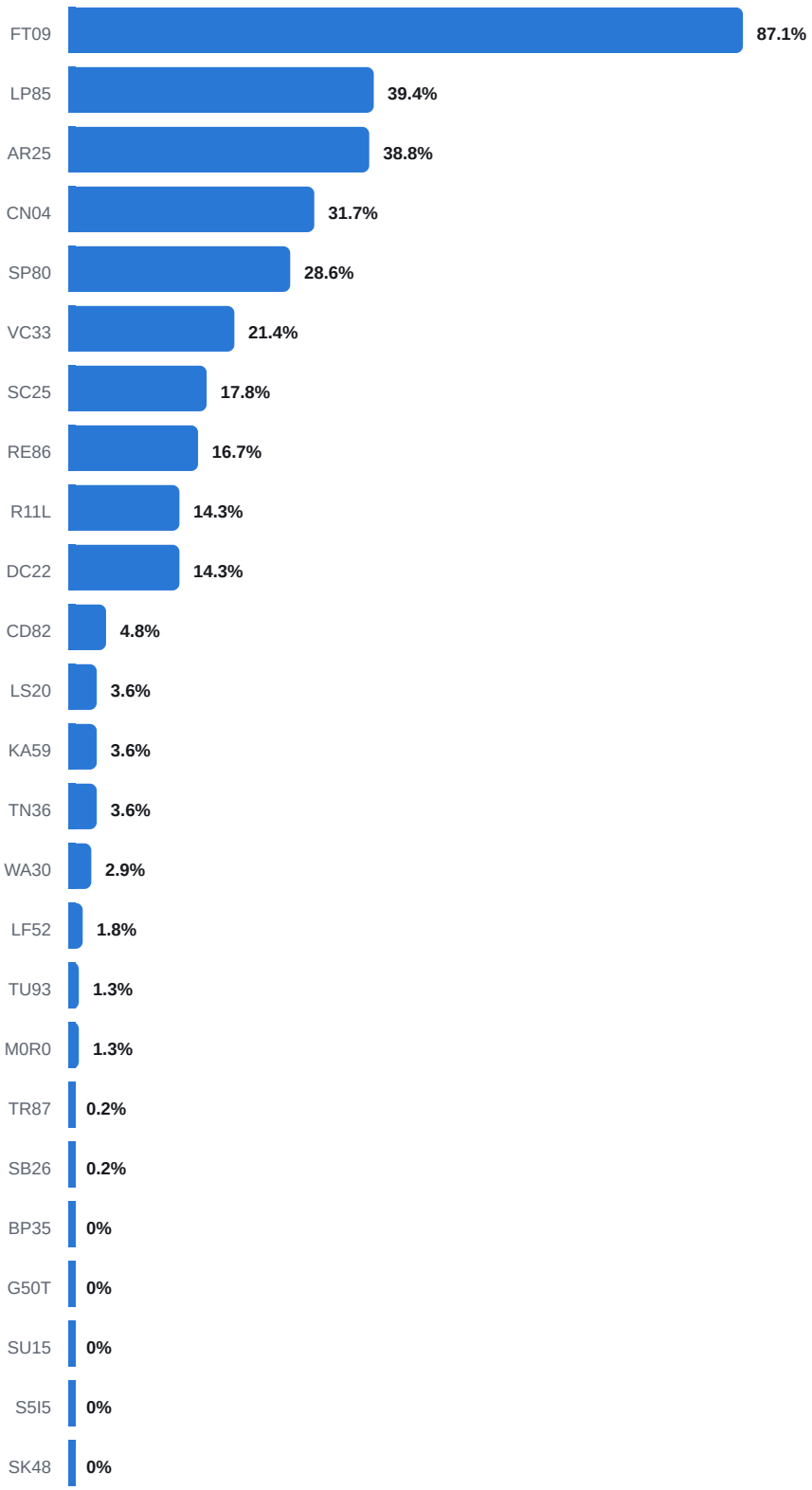
Figure 5. Which score may travel to which board — connections

From	To	Label
Public demo	Community board	permitted
Semi-private	Verified board	permitted
Public demo	Verified board	refused by the benchmark's own rule

One number in this story is not a single party's word. ARC Prize's own scorecard for the model publishes 13.33% on the public set, measured by ARC Prize, and its per-environment table sums to that figure across the 25 environments. The 38.3, by contrast, has one publisher, and no independent reproduction of it exists in the public record.

That per-environment table also shows how fragile a 25-environment mean is. One game supplies about a quarter of the total; ten of the twenty-five score at or below 1.8%, and five score exactly zero.

Figure 6. Where the 13.34% mean comes from: 25 environments, max reasoning effort



ARC Prize scorecard for GPT-5.6 Sol, read 2026-09-02. The 25 values sum to 333.4, a mean of 13.336%, which reproduces ARC Prize's published 13.33% and OpenAI's 13.3. FT09 alone supplies 26.1% of the total. OpenAI published no per-environment breakdown of its 38.3 run, so the same decomposition cannot be performed on the other side and it is not known how broadly that gain was spread.

▼ Table view

Figure 6. Where the 13.34% mean comes from: 25 environments, max reasoning effort

Environment	Score
FT09	87.1%
LP85	39.4%
AR25	38.8%
CN04	31.7%
SP80	28.6%
VC33	21.4%
SC25	17.8%
RE86	16.7%
R11L	14.3%
DC22	14.3%
CD82	4.8%
LS20	3.6%
KA59	3.6%
TN36	3.6%
WA30	2.9%
LF52	1.8%
TU93	1.3%
M0R0	1.3%
TR87	0.2%
SB26	0.2%
BP35	0%
G50T	0%
SU15	0%
S5I5	0%
SK48	0%

8. The replies, and the evidence that complicates them

ARC Prize answered publicly at 03:37 UTC on 30 July, calling the finding a real and useful result about harness design — which is not the same as admitting the score. Its stated reason for a deliberately thin runner is comparability: every provider receives the same observations, the same system prompt and the

same action limits, so that nobody can quietly tune the scaffolding to the test. It added that it was working with several laboratories, OpenAI among them, on how to incorporate provider-side state into verified testing while keeping it fair across providers.

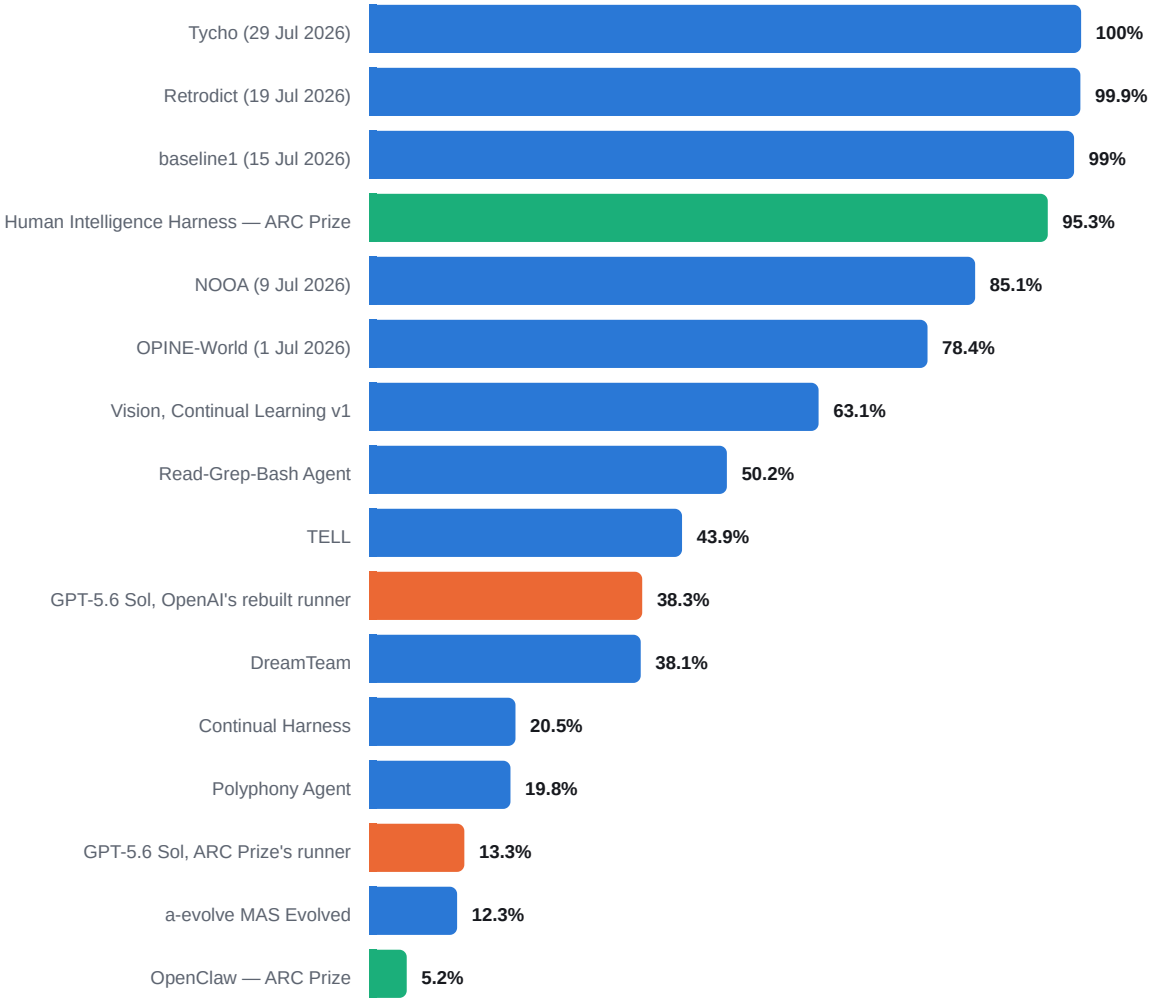
Four hours later François Chollet, who created ARC and co-founded ARC Prize, drew the line himself. A harness built specially for the benchmark, or containing knowledge of it, is not allowed; general-purpose settings available to every API customer are fine. He acknowledged the parity problem that leaves — different providers tested under different settings — and set a condition:

> My take is that this is fine as long as the settings and the cost are clearly reported.

He also recorded something the July note does not: that there had been "a lot of back and forth with OpenAI about how to best test their models, especially with regard to compaction". The discovery was not made in isolation.

The obvious conclusion at this point is that thicker software is better software. ARC Prize's own material refuses it three times over. It maintains a second, community leaderboard for precisely these harness-driven results, on the same 25 environments and the same metric, and the range there is very wide.

Figure 7. The same 25 environments, the same metric: fourteen systems and the two July figures



ARC Prize community leaderboard, read 2026-09-02, plus the two figures from OpenAI's note of 29 July 2026. Community entries are self-reported and lightly reviewed, and the two OpenAI figures were not submitted to that board; they are placed on the same axis because they are the same metric on the same 25 environments. The two rows marked as ARC Prize's own are its published harnesses.

▼ **Table view**

Figure 7. The same 25 environments, the same metric: fourteen systems and the two July figures

System	Score
Tycho (29 Jul 2026)	100%
Retrodict (19 Jul 2026)	99.9%
baseline1 (15 Jul 2026)	99%
Human Intelligence Harness — ARC Prize	95.3%
NOOA (9 Jul 2026)	85.1%
OPINE-World (1 Jul 2026)	78.4%
Vision, Continual Learning v1	63.1%
Read-Grep-Bash Agent	50.2%

System	Score
TELL	43.9%
GPT-5.6 Sol, OpenAI's rebuilt runner	38.3%
DreamTeam	38.1%
Continual Harness	20.5%
Polyphony Agent	19.8%
GPT-5.6 Sol, ARC Prize's runner	13.3%
a-evolve MAS Evolved	12.3%
OpenClaw — ARC Prize	5.2%

Three separate teams reached 99.0%, 99.9% and 100.0% on that set on 15, 19 and 29 July, the last of them on the day the note appeared — so the harness axis on this benchmark was already known to be very long, and 38.3 sits in the middle of it. More pointedly, ARC Prize's own agentic harness, given memory and the ability to execute code, scores 5.2%: below the thin runner it was built to improve on. Its account of the summer's first milestone prize reports the same phenomenon in the winning entry, a small open-weights model run locally:

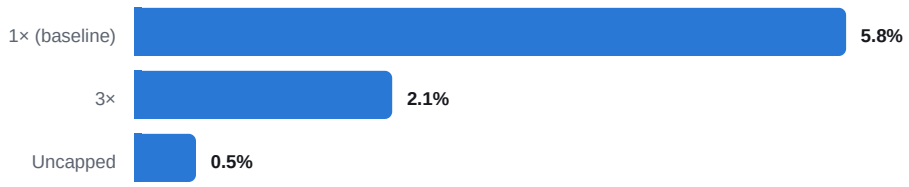
> Tufa Labs noted that, counterintuitively, hand-crafted tools actually hurt the model; letting it improvise worked better.

The second posture in this argument comes from a different company and a different problem. In a note of 24 March 2026 on harness design for software that runs for hours, an Anthropic engineer, Prithvi Rajasekaran, describes removing one scaffolding construct when a stronger model arrived, and keeping a planner and an evaluator because both continued to earn their cost. The note does not end on a finding, and is careful to say so:

> From this work, my conviction is that the space of interesting harness combinations doesn't shrink as models improve. Instead, it moves, and the interesting work for AI engineers is to keep finding the next novel combination.

That is one engineer's stated conviction after one project, not a company's finding. What the same company has published as a measurement is narrower and more useful. On 5 February 2026 Gian Segato held the model, the harness and the task set fixed and varied only how much machine resource each run was allowed.

Figure 8. Infrastructure error rate when only machine resources change



Anthropic, Quantifying infrastructure noise in agentic coding evals, 5 February 2026, on Terminal-Bench 2.0 with the model, harness and tasks held constant. The success-rate effect is smaller and does not track the error rate: 1× to 3× was within noise (p = 0.40), while 1× to uncapped moved success about 6 points (p < 0.01). The spread across the moderate range is just under 2 percentage points; 6 is the figure at the extremes.

▼ **Table view**

Figure 8. Infrastructure error rate when only machine resources change

Resource enforcement	Infrastructure error rate
1× (baseline)	5.8%
3×	2.1%
Uncapped	0.5%

The recommendation that follows from it is the practical one: treat a leaderboard gap of under about three percentage points with scepticism until the two setups are known to have matched. Academic work has since reached the same place from outside the industry. *Harness-Bench*, published on 27 May 2026 by Yilun Yao and colleagues, crossed harness configurations with model backends over 5,194 execution trajectories and concluded that "agent capability should be reported at the model-harness configuration level rather than attributed to the base model alone".

9. What is checkable, and when

Three things about this dispute can be re-read by anyone, on a date.

A column. ARC Prize said on 30 July that it was working out how to bring provider-side state into verified testing. The machine-readable file behind its leaderboard was regenerated on 1 September 2026 at 20:47 UTC; compared field by field against the copy the Internet Archive holds from 22 August, which the file itself stamps as generated on 21 August at 19:53 UTC, not one of its 27 rows differs in any field, and it still carries a cost column and no settings column. Chollet's condition was that settings and cost both be clearly reported, and the cost column already exists. The appearance of a settings column would be the whole argument resolving in public.

A prize, and a number already moving. The second and final ARC-AGI-3 milestone prize closes on 30 September 2026, and that track runs without internet access, so no commercial API-based system can enter it. Its high score reached 4.58% on 24 August, which ARC Prize said followed one team open-sourcing its solution, after which, in its own words, "others quickly built on top of it and pushed the scores higher". By 31 August it stood at 7.51%, posted by a different entrant — a rise of about two-thirds in a week, announced in a single line with no explanation and no description of what produced it.

A direction of travel. No frontier laboratory has published a successor to the March harness note: Anthropic's engineering index has nothing newer than 25 May 2026, and OpenAI has published nothing on harnesses since 29 July. One academic harness paper has appeared since — *openJiuwen*, 28 August 2026 — and every element of its contribution is an addition rather than a removal: composable rails, delegated sub-agents, runtime-adaptive control. Its claimed margins over the leaderboard, 3.4 and 3.39 percentage points, sit barely above the three-point noise floor another of the sources here proposed six months earlier.

10. The limits of the record

Five things the available evidence will not support.

- **A capability multiple.** The ratio 38.3/13.3 is 2.88 on one constructed metric on one set of 25 environments under two configurations; no broader multiplier for capability, usefulness or value follows from it.
- **A cost saving.** The six-fold figure counts output tokens per game at one reasoning setting, and no dollar figure for either run has been published.
- **A clean two-variable experiment.** At least four things differ between the configurations, one of them a change of API and one a change of truncation unit that only the interested party has assessed.
- **Independent confirmation of the improvement.** ARC Prize's 13.33% corroborates the baseline. No independent reproduction of the 38.3 has been published, and ARC Prize's reply characterises OpenAI's evidence as internal testing rather than as a run of its own.
- **A general rule that thicker harnesses win.** On this benchmark the harness axis runs from 5.2% to 100% on a fixed set, and one of the lowest entries is a fully agentic harness. Work on coding benchmarks published on 8 June 2026 found harness choice moving tokens per solved task by up to 40× while paired within-model pass-rate differences stayed between 0 and 8 percentage points — and its authors report that the confidence intervals on those differences include zero for every gap but the largest. How much the harness is worth appears to depend on whether the task requires the system to remember.

11. What to keep

Any benchmark on which a system must act repeatedly and carry something forward between actions is partly a measurement of the software around the model. Not because the benchmark is corrupt, but because the boundary is drawn by whoever is measuring, and the number belongs to everything inside it.

Two questions cost nothing and recover most of the picture. **What exactly was scored** — which benchmark, which of its sets, at which setting? And **what changed** — because the trained object can be identical while the arrangement around it is not.

SPEC answered both by requiring a configuration sheet, in 1988, for a machine that could not talk back. The July episode is what the same question looks like when the component under test produces sentences, and the paperwork has not caught up.

12. Sources

Source	Date	Location
OpenAI (Ilan Bigio, Ted Sanders), *How enabling two settings tripled our scores on the ARC-AGI-3 benchmark*, and the ten datapoints embedded in its chart	29 Jul 2026; chart data extracted 2 Sep 2026	openai.com/index/how-two-settings-tripled-our-arc-agi-3-scores/
ARC Prize Foundation, *ARC-AGI-3: A New Challenge for Frontier Agentic Intelligence*	arXiv v1 24 Mar 2026, **v2 17 Apr 2026**	arXiv 2603.24621
ARC Prize (@arcprize), public reply on X	30 Jul 2026, 03:37 UTC	x.com/arcprize/status/2082672003765670160
François Chollet (@fchollet), post on X	30 Jul 2026, 07:37 UTC	x.com/fchollet/status/2082732210436575669
ARC Prize, GPT-5.6 Sol scorecard: 13.33% public, 7.78% semi-private, per-environment table	read 2 Sep 2026	arcprize.org/results/openai-gpt-5-6-sol
ARC Prize, community leaderboard	read 2 Sep 2026	arcprize.org/leaderboard/community
ARC Prize, verified leaderboard data, 27 rows	generated 1 Sep 2026, 20:47 UTC	arcprize.org/media/data/leaderboard/v3.json
ARC Prize, *ARC Prize 2026: ARC-AGI-3 Milestone Prize #1*	6 Jul 2026	arcprize.org/blog/arc-prize-2026-milestone-1
ARC Prize, *Verified Testing Policy*; and the 2026 competition key dates and Kaggle conditions	both read 2 Sep 2026	arcprize.org/policy; arcprize.org/competitions/2026
Anthropic (Prithvi Rajasekaran), *Harness design for long-running application development*	24 Mar 2026	anthropic.com/engineering/harness-design-long-running-apps
Anthropic (Gian Segato), *Quantifying infrastructure noise in agentic coding evals*	5 Feb 2026	anthropic.com/engineering/infrastructure-noise
Standard Performance Evaluation Corporation, *SPEC CPU 2017 Run and Reporting Rules*, rule 1.2.2; and SPEC's own account of its 1988 founding	read 2 Sep 2026	spec.org/cpu2017/Docs/runrules.html; spec.org/spec/
Text REtrieval Conference overview, NIST	read 2 Sep 2026	trec.nist.gov/overview.html
Yao, Tan, Liu, Li, Wang et al., *Harness-Bench: Measuring Harness Effects across Models in Realistic Agent Workflows*	27 May 2026	arXiv 2605.27922
Vats and Golev, *The Scaffold Effect in Coding Agents*	8 Jun 2026	arXiv 2607.22585
openJiuwen Team et al., *openJiuwen: Beyond Static Harnesses for Long-Horizon Coding Agents*	28 Aug 2026	arXiv 2608.27969