

How good is an agent harness when the model is held constant?

Six agent stacks, the same 22 tasks, the same model: Claude Sonnet 5.5. We measured what each stack adds or costs. HarnessX, DataGOL's harness, is highlighted throughout.

What the benchmark does

One model, one task set, one set of graders. Only the harness changes.

Every stack gets **the same system prompts, tools and user prompts**. A recording proxy sits between each stack and the API and logs every call, so cost, latency, retries and tokens come from the wire and not from what a framework says about itself. Tasks are graded by deterministic checks, plus an Opus judge for the one open-ended writing task.

TIER	What it tests	What the agent has to do	TASKS
A	Tool use	Chain, fan out and paginate calls to simple tools, and keep state across them	5
B	Coding	Fix a bug, add a feature, refactor, debug, implement from a spec, in a real sandbox	5
C	Corpus research	Read a 40-document corpus, join facts across files, spot contradictions, write a brief	3
D	Long context	Cross-reference 30 modules or 120 patches without losing the thread	2
E	Strict JSON	Return exactly the requested JSON from messy inputs and tool output	3
F	Adversarial tools	Transient errors, huge outputs, misleading results, prompt injection	4

6

stacks: HarnessX, Deep Agents, CrewAI, pi, TrueForge and a raw SDK baseline

22

tasks in six tiers, 1 to 3 repetitions each

456

graded runs: 360 capability and 96 fault-injection

\$34.4

API spend across those runs

TWO PROFILES

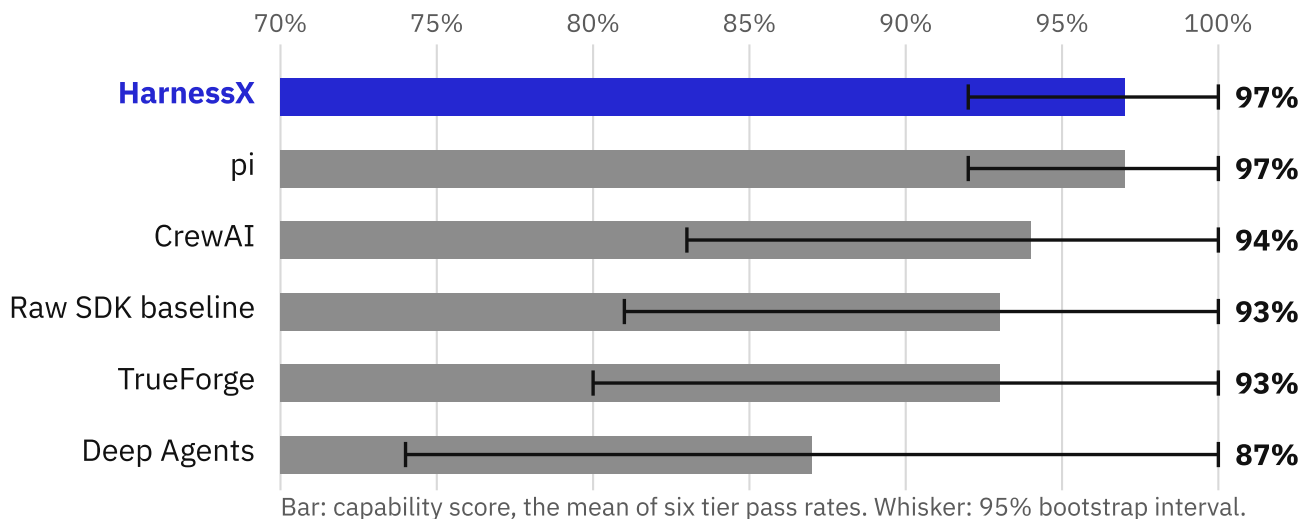
Tuned is how a team would deploy a stack: explicit caps, retries, caching. **Defaults** is the framework as shipped. Capability numbers use tuned.

A FAULT SUITE

A proxy injects rate limits, 500s, overloads and timeouts. Each stack is scored on whether it recovers, fails cleanly, answers wrongly, hangs or crashes. This suite runs on Claude Haiku 4.5, since it tests the retry path and not the model.

Capability: HarnessX ties for first

HarnessX scored **97%**, tied with pi for first. The intervals overlap heavily, so ranks 1 to 5 cannot be separated statistically. Deep Agents (87%) is the lowest.



WHERE THE POINTS ARE WON AND LOST

	A Tools	B Coding	C Research	D Long ctx	E JSON	F Adversarial
HarnessX	100%	100%	83%	100%	100%	100%
pi	100%	100%	83%	100%	100%	100%
CrewAI	100%	100%	67%	100%	100%	100%
Raw SDK baseline	93%	100%	67%	100%	100%	100%
TrueForge	100%	100%	67%	100%	89%	100%
Deep Agents	100%	100%	83%	50%	89%	100%

Pass rate per tier. Tier runs: A 15, B 10, C 6, D 4, E 9, F 8. One run moves C by 17 points and D by 25

STRENGTHS

- **100% on five of six tiers**, including all 15 tool-use runs and all long-context runs.
- Only HarnessX and pi had no tier below 83%.

GAP

- **Corpus research (C): 83%** (5 of 6), the one miss on the contradictions task (c3). It finds every planted contradiction but lists extras the grader counts as false positives. The corpus looks to hold other real inconsistencies, so c3 is under audit.
- The baseline, CrewAI and TrueForge scored 67% in C.

HarnessX, task by task

Tuned profile. Each tile is one task: runs passed out of runs.



21 of 22 tasks pass every run, including all 15 tool-use runs.

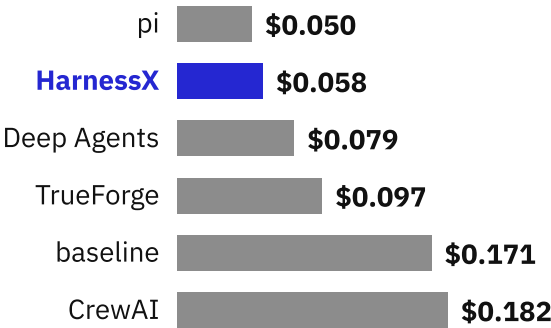
The only miss is c3 (contradictions), wrong in 1 of 2 runs: the model finds the planted pairs but also lists extras that the grader counts as false positives.

HARNESSX AGAINST THE RAW SDK BASELINE

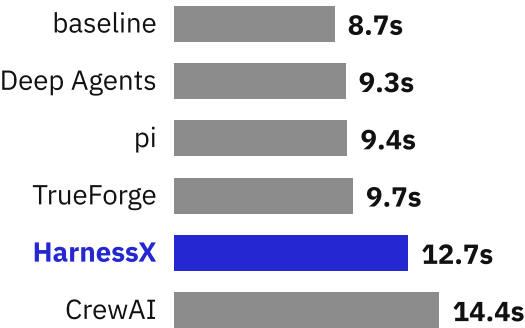
MEASURE (SONNET, TUNED)	HARNESSX	RAW BASELINE
Capability score	97%	93%
Cost per success	\$0.058	\$0.171
Median cost per run	\$0.0309	\$0.0322
Median wall time per run	12.7 s	8.7 s
Framework overhead	1%	1%

COST AND SPEED

Cost per success (\$)

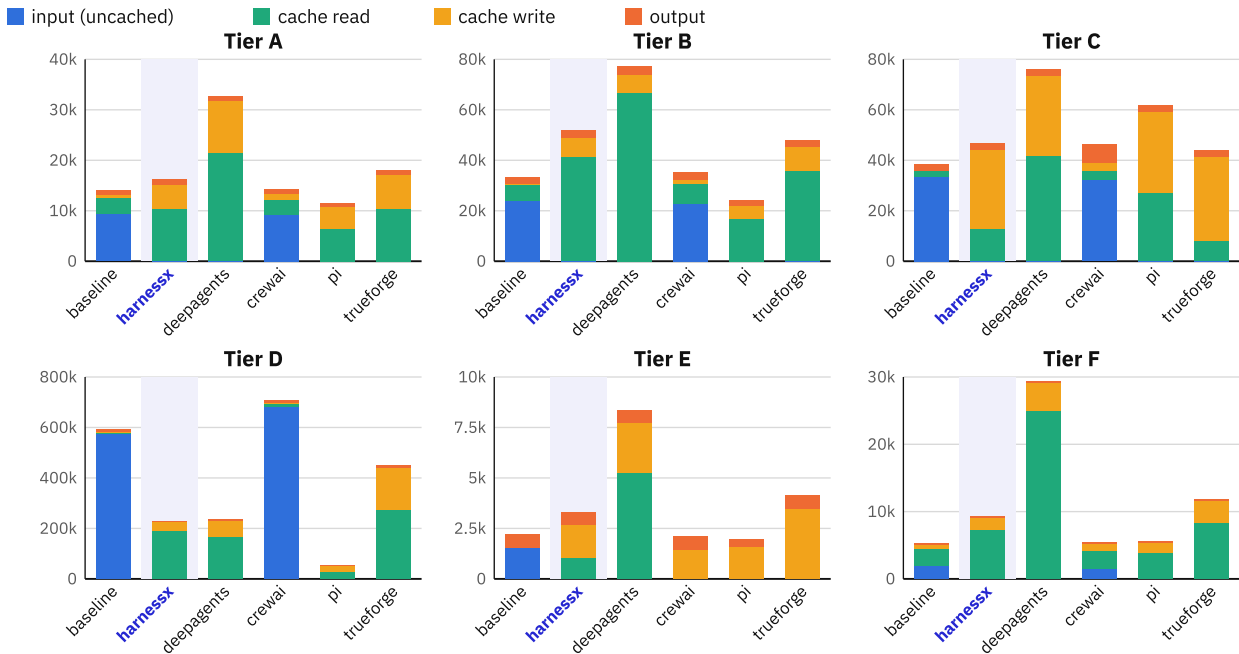


Median wall time per run (s)

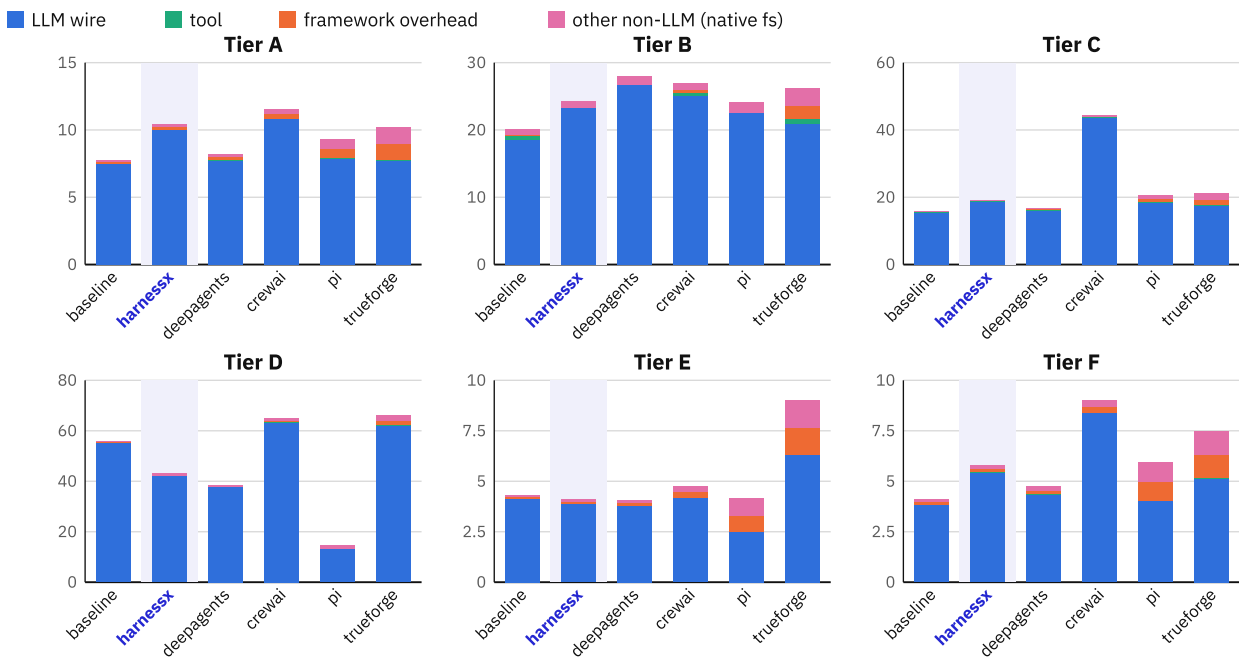


Second cheapest per success, about a third of the baseline and CrewAI. Framework overhead is 1%, tied lowest, so the wall-time gap is model time, not framework time: on short tasks each model call takes longer, partly from HarnessX's planning turns and a larger prompt.

Median tokens per run, by tier

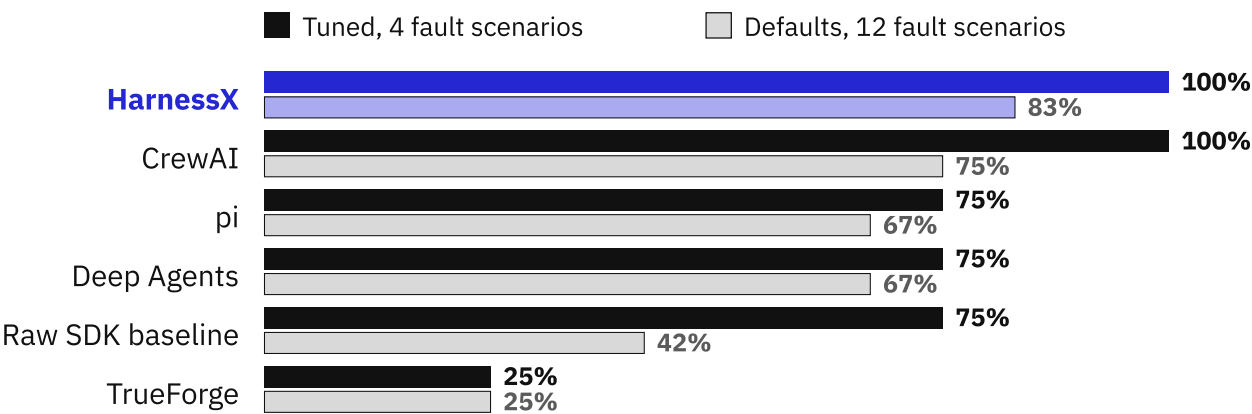


Wall-clock decomposition, by tier



Reliability: HarnessX recovered the most faults

Share of fault scenarios where the run still produced a correct answer. This suite runs on Claude Haiku 4.5.



READING THE CHART

Tuned, HarnessX recovered all 4 fault scenarios, tied with CrewAI. As shipped, it recovered 10 of 12 and failed cleanly on the other 2, the best of the six. TrueForge recovered 3 of 12 on defaults and had a hang.

Takeaways

WHAT HELD UP

- Capability tied for first at 97%.
- Second-lowest cost per success and 1% framework overhead.
- Best or tied-best fault recovery on both profiles.
- 100% on five of six tiers.

WHAT TO WATCH

- Wall time: 12.7 s median against 8.7 s for the baseline, mostly model time.
- The c3 contradictions miss, under audit.