FINAL REPORT · SINGLE SUITE · 7 TESTS · 27 AUGUST 2026

BENCH_V93C

Comparative benchmark — OpenFreedom V.9.3 · DYNAMIC THINKING
OpenClaw (OC) · **OpenFreedom (OF)** — single model deepseek-v4-flash

OpenClaw 2026.7.1-2 | OF V.9.3 · F4 high | OF V.9.3 · F4 dinamico

27 August 2026

Purpose of the test

1. Verify the **overall endurance** of OpenFreedom vs OpenClaw: the suite covers real creation, reasoning and repair tasks — and OC **completes only 4 out of 7 tasks**.
2. Measure how the new **V.9.x OF**, with the **dynamic reasoning** system (F4 reasoning level chosen per-task from the gate complexity score, after the F1 plan), **further cuts OF's own times and costs** at equal quality.

How the test was run

The **single suite** of 7 tests from the previous benchmark was reused (`prompts-unica.json`): prompts written as a real user would, with no hints. The 7 cases cover: direct knowledge, admission-test math, financial strategy, building a web app, a complete offline e-commerce, an artisan CRM, and repairing a ~100 KB source file with 5 bugs.

Procedure: for each test the OC platform ran as baseline; for OF the two configurations (**F4 high** and **F4 dynamic**) ran as **adjacent interleaved pairs** (alternating order across tests) so cache and load conditions hit both configurations equally. Before each run the state was **fully reset** (`reset-cache.sh`): caches and sessions, OF dialog/context, stellar memory restored to the atavic base (13 nodes, zero memories from previous runs), test artifacts on disk, service restarts. For each test × configuration: exact prompt, full response, usage (billed tokens) and generated files were saved.

Machine and installations

- **Same machine** for all runs: Linux PC (8 GB RAM), each task launched one at a time to avoid bottlenecks.
- **Clean caches** before every run: sessions, dialog, stellar memory and on-disk artifacts cleared.
- **VANILLA installations** — official releases, no custom changes for the test: OpenClaw 2026.7.1-2 (official gateway, direct injection), OpenFreedom V.9.3 (official webui).
- **Single model:** deepseek-v4-flash for all platforms.

- **Single pricing (billed tokens, $/M):** new input 0.14 · output 0.28 · cache 0.028 — identical for all; *new_in* = total input − cache.

The test is repeatable

- Suite and runner versioned and reproducible: `prompts/prompts-unica.json`, `runner/run_task.py`, `reset-cache.sh`.
- For T7 the master file (`gestionale.py`, 97,987 bytes, md5 `3f0cfaef…`) is never touched: each platform works on a copy refreshed from the master, integrity verified (md5 pre/post) + 14-test suite.

T1 — Conoscenza diretta

Prompt (starting text, no hints)

> *Mi sai dire chi era il dio Giano per i romani?*

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	19.9 s	1	5,980	672	8,448	$0.0013	
OF V.9.3 · F4 high	14.8 s	2	1,058	1,258	13,312	$0.0009	
OF V.9.3 · F4 dinamico	6.1 s	2	8,460	280	5,376	$0.0014	

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v0 · low**

T2 — Matematica da quiz di ammissione

Prompt (starting text, no hints)

> *Mi stai preparando per il test di ammissione all'università e sono bloccato su un esercizio. Un rubinetto riempie un serbatoio in 4 ore, un secondo rubinetto lo riempie in 6 ore e lo scarico lo svuota in 3 ore. Se apro tutti e tre contemporaneamente, in quanto tempo si riempie il serbatoio?*

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	18.0 s	1	6,057	650	8,448	$0.0013	
OF V.9.3 · F4 high	8.0 s	2	1,017	622	13,056	$0.0007	
OF V.9.3 · F4 dinamico	10.6 s	2	5,246	1,032	8,832	$0.0013	

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v4 · low**

T3 — Strategia finanziaria su dati personali

Prompt (starting text, no hints)

Ho 34 anni e guadagno 2.300 euro netti al mese. Porto avanti un mutuo sulla casa con una rata di 780 euro al mese (mancano 18 anni), un prestito per l'auto da 210 euro al mese (mancano 2 anni) e un finanziamento per l'arredamento da 95 euro al mese (mancano 3 anni). Ho 6.500 euro da parte. Come mi conviene organizzare le mie finanze? Quanto mi resta davvero ogni mese, ha senso estinguere prima qualche debito e come potrei iniziare a mettere da parte qualcosa in modo prudente?

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	59.4 s	2	0	4,764	34,688	$0.0023	
OF V.9.3 · F4 high	19.5 s	2	1,356	1,755	12,928	$0.0010	
OF V.9.3 · F4 dinamico	18.8 s	2	1,138	1,645	13,184	$0.0010	

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v9 · low**

T4 — Web app: la lista della spesa

Prompt (starting text, no hints)

Mi serve una web app per gestire la spesa di casa: aggiungo i prodotti che compro con prezzo e quantità, modifico quelli esistenti e li elimino. I dati devono restare salvati nel browser. La web app deve essere in un singolo file html con tutto incorporato (css e javascript), senza dipendenze esterne e deve funzionare offline.

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	309.7 s	15	0	27,871	620,288	$0.0252	
OF V.9.3 · F4 high	26.2 s	2	11,820	3,179	5,120	$0.0027	
OF V.9.3 · F4 dinamico	40.2 s	3	1,184	5,128	21,376	$0.0022	

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v9 · low**

T5 — E-commerce offline completo: negozio di cappelli

Prompt (starting text, no hints)

> *Vorrei aprire un negozio online di cappelli. Mi serve un sito completo che funzioni tutto: catalogo con foto, carrello, checkout, pagine prodotto, tutto salvato in locale senza server. Deve essere una web app in un unico file html, con design moderno, bottoni funzionanti e navigazione tra le pagine.*

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	310.1 s	17	0	45,321	869,760	$0.0370	
OF V.9.3 · F4 high	49.8 s	2	12,356	7,696	8,064	$0.0041	
OF V.9.3 · F4 dinamico	57.4 s	2	13,888	8,407	6,656	$0.0045	

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v16 · medium**

T6 — CRM webapp per piccola ditta artigiana

Prompt (starting text, no hints)

> *Gestisco una piccola impresa edile artigiana e mi serve un programma per tenere in ordine clienti, preventivi, materiali e fatture. Deve avere un menù semplice, salvare i dati su file e stampare i preventivi. Meglio una sola pagina web che contiene tutto, con grafica professionale e un design moderno.*

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	255.5 s	6	0	30,106	264,832	$0.0158	
OF V.9.3 · F4 high	60.2 s	2	8,847	10,521	8,192	$0.0044	
OF V.9.3 · F4 dinamico	48.9 s	2	1,331	8,029	12,800	$0.0028	

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v14 · medium**

T7 — Analisi e riparazione di gestionale.py (~100 KB, 5 bug nascosti)

Prompt (starting text, no hints)

Ciao, ti passo il file del programma che uso in magazzino per i prodotti e le fatture: /tmp/bench20/ripara/gestionale.py. Non funziona più bene come prima, alcuni calcoli sbagliano e i test falliscono. Sistemalo per favore mantenendo la struttura.

Comparison table

Piattaforma / config	Tempo	LLM	Input	Output	Cache	Costo	Esito
OpenClaw	106.5 s	16	0	10,411	1,082,112	$0.0332	14/14
OF V.9.3 · F4 high	70.2 s	3	44,213	8,689	54,528	$0.0101	14/14
OF V.9.3 · F4 dinamico	36.6 s	2	255	4,482	54,784	$0.0028	14/14

Gate complexity score and F4 level chosen by DYNAMIC THINKING: **v26 · high**

Overall verdict

Test	OC	OF V.9.3 · high	OF V.9.3 · dinamico
T1 · Knowledge	×	✓	✓
T2 · Math	×	✓	✓
T3 · Finance	✓	✓	✓
T4 · Web app	✓	✓	✓
T5 · E-commerce	✓	✓	✓
T6 · CRM	✓	✓	✓
T7 · Repair 100 KB	14/14	14/14	14/14
Total	**4/7**	**7/7**	**7/7**

OpenFreedom is the only platform that solves all creation tasks (T4–T6) and the repair (T7, 14/14 tests, master intact). **OpenClaw** passes 4 out of 7: on creation tasks it burns 15–17 LLM calls and ~5 minutes per test without delivering a file. On T7 (code repair) both finish 14/14, but OF takes 2–3 LLM calls vs OC's 16.

The effect of DYNAMIC THINKING (V.9.x)

The **dynamic reasoning** chooses per-task the F4 reasoning level from the **gate deterministic complexity score** (0-50, computed after the F1 plan): trivial → **low**, work tasks → **medium**, complex repairs → **high**. In the suite: T1–T4 low (score 0–9), T5/T6 medium (score 14–16), T7 high (score 26).

Metric (suite T1–T7)	OC	OF · F4 high	OF · F4 dinamico	dynamic vs high	dynamic vs OC
Total time	1079 s	249 s	219 s	**−12,1%**	**−79,7%**
Total cost	$0,1161	$0,0240	$0,0160	**−33,3%**	**−86,2%**
LLM calls	58	15	15	0%	−74%
Tests passed	4/7	7/7	7/7	—	—

Reading: dynamic cuts **33% of cost** and **12% of time** vs the same version with F4 fixed at high, at equal quality (7/7, T7 14/14 both). The gain is on trivial tasks (T1–T3 go low) while complex tasks stay protected (T7 stays high). vs OC, OF-dynamic costs **−86%** and takes **−80%** of the time, completing 7/7 vs 4/7.

Technical note — T7: why times halve at the same high reasoning

In T7 both OF configurations used F4 **high** (dynamic scored 26/50 → high): the reasoning level is identical. The time (70.2 s vs 36.6 s) and cost ($0.0101 vs $0.0028) gap has two causes: **(1)** the high run took **3 LLM calls** (one extra verification/repair round, ~30 s) while the dynamic run finished in **2**; **(2)** the dynamic run, executed right after in the interleaved pair, benefited from the provider **prefix cache** (new input 255 vs 44,213 tokens), making input almost free. It is not the reasoning: it is round count + cache. In the suite total, the interleaved order distributes this effect equally between the two configurations.

Results clarification

For OC, the outcomes on creation tasks (T4–T6) are recorded with the same method used identically for all platforms (check of expected strings on response + generated files); OC delivered no capturable file, after 6–17 LLM calls and ~5 minutes per test. The usage and time burned are real and measured.

References

- **DeepSeek API pricing** — `api-docs.deepseek.com/quick_start/pricing`: per-token rates used in the cost calculation (0.14 / 0.28 / 0.028 $/M).
- **Anthropic — "Building Effective Agents"** — `anthropic.com/engineering/building-effective-agents`: deterministic workflows often outperform autonomous agents — architectural basis of the OF gate.

Disclaimer and Legal Notes

1. Informational Purpose and Independence

The data, results and performance analyses in this comparison table have an exclusively informational and scientific purpose. This benchmark was conducted in full independence. There is no affiliation, sponsorship, partnership or commercial agreement between this site and the owners or developers of the OpenClaw project or the cited LLMs. All trademarks and product

names belong to their rightful owners and are mentioned here solely for identification and fair scientific comparison.

2. Time and Version Limits

AI agent performance depends directly on the software and models used. Tests were run in August 2026 using the vanilla (standard, unmodified) releases of the respective software publicly available at that date. As these are fast-evolving open source technologies, subsequent updates may vary, improve or alter the results shown here.

3. Terms of Use and Reproducibility

The published results reflect exclusively the performance obtained within the isolated hardware and software environment described in the test methodology (same machine, same underlying LLM, same prompts, sequential execution, cleared caches and vanilla installations). Although the test was performed in good faith and with maximum objectivity criteria, performance in real production environments or with different configurations may vary. Users are invited to verify the data independently by replicating the benchmark using the parameters indicated in the technical documentation.