REPORT FINALE · SUITE UNICA · 7 TEST · 27 AGOSTO 2026

BENCH_V93C

Benchmark comparativo — OpenFreedom V.9.3 · THINKING DINAMICO
OpenClaw (OC) · **OpenFreedom (OF)** — modello unico deepseek-v4-flash

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

27 agosto 2026

Scopo della prova

1. Verificare la **tenuta complessiva** di OpenFreedom rispetto a OpenClaw: la suite copre compiti reali di creazione, ragionamento e riparazione — e OC **non porta a termine 4 task su 7**.
2. Misurare come la nuova versione **V.9.x di OF**, con il sistema di **reasoning dinamico** (il livello di ragionamento di F4 viene scelto per-task dal voto di complessità del gate, dopo il piano di F1), **abbatta ulteriormente i tempi e i costi di OF stesso** a parità di qualità.

Come si è svolta la prova

È stata riutilizzata la **suite unica** di 7 test del benchmark precedente (`prompts-unica.json`): prompt scritti come li scriverebbe un utente reale, senza alcun aiutino. I 7 casi coprono: conoscenza diretta, matematica da quiz di ammissione, strategia finanziaria, costruzione di una web app, di un e-commerce offline completo, di un CRM artigianale e la riparazione di un file sorgente di ~100 KB con 5 bug.

Svolgimento: per ogni test la piattaforma OC è stata eseguita come baseline; per OF le due configurazioni (**F4 high** e **F4 dinamico**) sono state eseguite **a coppie adiacenti interleaved** (ordine alternato tra i test) in modo che cache e condizioni di carico colpiscano allo stesso modo entrambe le configurazioni. Prima di ogni run lo stato è stato **azzerato completamente** (`reset-cache.sh`): cache e sessioni, dialogo/contesto OF, memoria stellare riportata alla base atavica (13 nodi, zero ricordi dei run precedenti), artefatti di prova su disco, riavvio dei servizi. Per ogni test × configurazione sono stati salvati: prompt esatto, risposta completa, consumi (token fatturati) e file generati.

Macchina e installazioni

- **Stessa macchina** per tutti i run: PC Linux (RAM 8 GB), ogni task lanciato uno alla volta per evitare colli di bottiglia.
- **Cache pulite** prima di ogni run: sessioni, dialogo, memoria stellare e artefatti su disco azzerati.

- **Installazioni VANILLA** — release ufficiali, nessuna modifica custom per la prova: OpenClaw 2026.7.1-2 (gateway ufficiale, iniezione diretta), OpenFreedom V.9.3 (webui ufficiale).
- **Modello unico:** deepseek-v4-flash per tutte le piattaforme.
- **Listino unico (token fatturati, $/milione):** input nuovo 0,14 · output 0,28 · cache 0,028 — identico per tutti; *new_in* = input totale − cache.

La prova è ripetibile

- Suite e runner versionati e riproducibili: `prompts/prompts-unica.json`, `runner/run_task.py`, `reset-cache.sh`.
- Per il T7 il file master (`gestionale.py`, 97.987 byte, md5 `3f0cfaef…`) non viene mai toccato: ogni piattaforma lavora su una copia rinfrescata dal master e l'integrità è verificata (md5 pre/post) + suite di 14 test.

T1 — Conoscenza diretta

Prompt (testo di partenza, nessun hint)

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

Tabella comparativa

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	

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v0 · low**

T2 — Matematica da quiz di ammissione

Prompt (testo di partenza, nessun hint)

> *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?*

Tabella comparativa

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	

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v4 · low**

T3 — Strategia finanziaria su dati personali

Prompt (testo di partenza, nessun hint)

> *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?*

Tabella comparativa

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	

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v9 · low**

T4 — Web app: la lista della spesa

Prompt (testo di partenza, nessun hint)

> *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.*

Tabella comparativa

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	

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v9 · low**

T5 — E-commerce offline completo: negozio di cappelli

Prompt (testo di partenza, nessun hint)

> *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.*

Tabella comparativa

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	

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v16 · medium**

T6 — CRM webapp per piccola ditta artigiana

Prompt (testo di partenza, nessun hint)

> *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.*

Tabella comparativa

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	

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v14 · medium**

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

Prompt (testo di partenza, nessun hint)

> *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.*

Tabella comparativa

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

Voto di complessità del gate e livello F4 scelto dal THINKING DINAMICO: **v26 · high**

Verdetto complessivo

Test	OC	OF V.9.3 · high	OF V.9.3 · dinamico
T1 · Conoscenza	×		
T2 · Matematica	×		
T3 · Finanza			
T4 · Web app			
T5 · E-commerce			
T6 · CRM			
T7 · Riparazione 100 KB	14/14	14/14	14/14
Totale	**4/7**	**7/7**	**7/7**

OpenFreedom è l'unica piattaforma che risolve tutti i task di creazione (T4–T6) e la riparazione (T7, 14/14 test, master intatto). **OpenClaw** supera 4 test su 7: sui task di creazione brucia 15–17 chiamate LLM e ~5 minuti per test senza consegnare un file. Sul T7 (riparazione di codice) entrambe chiudono 14/14, ma OF impiega 2–3 chiamate LLM contro le 16 di OC.

L'effetto del THINKING DINAMICO (V.9.x)

Il **reasoning dinamico** sceglie per ogni task il livello di ragionamento di F4 dal **voto di complessità deterministico del gate** (0-50, calcolato dopo il piano di F1): banali → **low**, task di lavoro → **medium**, riparazioni complesse → **high**. Nella suite: T1–T4 low (voto 0–9), T5/T6 medium (voto 14–16), T7 high (voto 26).

Metrica (suite T1–T7)	OC	OF · F4 high	OF · F4 dinamico	dinamico vs high	dinamico vs OC
Tempo totale	1079 s	249 s	219 s	**−12,1%**	**−79,7%**
Costo totale	$0,1161	$0,0240	$0,0160	**−33,3%**	**−86,2%**
Chiamate LLM	58	15	15	0%	−74%
Test superati	4/7	7/7	7/7	—	—

Lettura: il dinamico abbatte del **33% il costo** e del **12% il tempo** di OF rispetto alla stessa versione con F4 fisso ad high, a parità di qualità (7/7, T7 14/14 in entrambe). Il guadagno sta nei task banali (T1–T3 vanno a low) mentre i task complessi restano protetti (T7 resta ad high). Rispetto a OC, OF-dinamico costa **−86%** e impiega **−80%** del tempo, completando 7/7 contro 4/7.

Nota tecnica — T7: perché i tempi si dimezzano a parità di reasoning high

Nel T7 entrambe le configurazioni OF hanno usato F4 **high** (il dinamico ha votato 26/50 → high): il livello di ragionamento è identico. La differenza di tempo (70,2 s vs 36,6 s) e costo ($0,0101 vs $0,0028) ha due cause: **(1)** la run high ha impiegato **3 chiamate LLM** (un round di verifica/riparazione in più, ~30 s) mentre la run dinamica ha chiuso in **2**; **(2)** la run dinamica, eseguita subito dopo nella coppia interleaved, ha beneficiato della **prefix cache** del provider (input nuovo 255 vs 44.213 token) che rende l'input quasi gratuito. Non è il reasoning: è numero di round + cache. Nel totale della suite l'ordine interleaved distribuisce questo effetto equamente tra le due configurazioni.

Chiarimento esiti

Per OC gli esiti dei task di creazione (T4–T6) sono registrati con lo stesso metodo identico per tutte le piattaforme (verifica delle stringhe attese su risposta + file generati); OC non ha consegnato alcun file catturabile, dopo 6–17 chiamate LLM e ~5 minuti per test. I consumi e i tempi bruciati sono reali e misurati.

Riferimenti

- **DeepSeek API pricing** — `api-docs.deepseek.com/quick_start/pricing`: tariffe per token usate nel calcolo dei costi (0,14 / 0,28 / 0,028 $/M).
- **Anthropic — "Building Effective Agents"** — `anthropic.com/engineering/building-effective-agents`: i workflow deterministici spesso superano gli agenti autonomi — base architetturale del gate di OF.

Clausola di Esclusione di Responsabilità e Note Legali (Disclaimer)

1. Scopo Informativo e Indipendenza

I dati, i risultati e le analisi prestazionali riportati in questa tabella comparativa hanno uno scopo esclusivamente informativo e scientifico. Questo benchmark è stato condotto in totale indipendenza. Non esiste alcuna affiliazione, sponsorizzazione, partnership o accordo commerciale tra questo sito e i proprietari o sviluppatori dei progetti OpenClaw o dei rispettivi LLM citati. Tutti i marchi e i nomi dei prodotti appartengono ai legittimi proprietari e vengono qui menzionati unicamente a fini identificativi e di leale confronto scientifico.

2. Limiti Temporali e di Versione

Le prestazioni degli agenti AI dipendono direttamente dal software e dai modelli utilizzati. I test sono stati eseguiti in data Agosto 2026 utilizzando le versioni vanilla (standard e non modificate) dei rispettivi software disponibili pubblicamente a tale data. Trattandosi di tecnologie open source in rapida evoluzione, i successivi aggiornamenti potrebbero variare, migliorare o alterare i risultati qui mostrati.

3. Condizioni d'Uso e Riproducibilità

I risultati pubblicati riflettono esclusivamente le prestazioni ottenute all'interno dell'ambiente hardware e software isolato descritto nella metodologia di test (stessa macchina, stesso LLM di fondo, medesimi prompt, esecuzione sequenziale, cache azzerate e installazioni vanilla). Sebbene il test sia stato eseguito in buona fede e con criteri di massima oggettività, le prestazioni in ambienti di produzione reali o con configurazioni differenti potrebbero variare. L'utente è invitato a verificare autonomamente i dati replicando il benchmark tramite i parametri indicati nella documentazione tecnica.