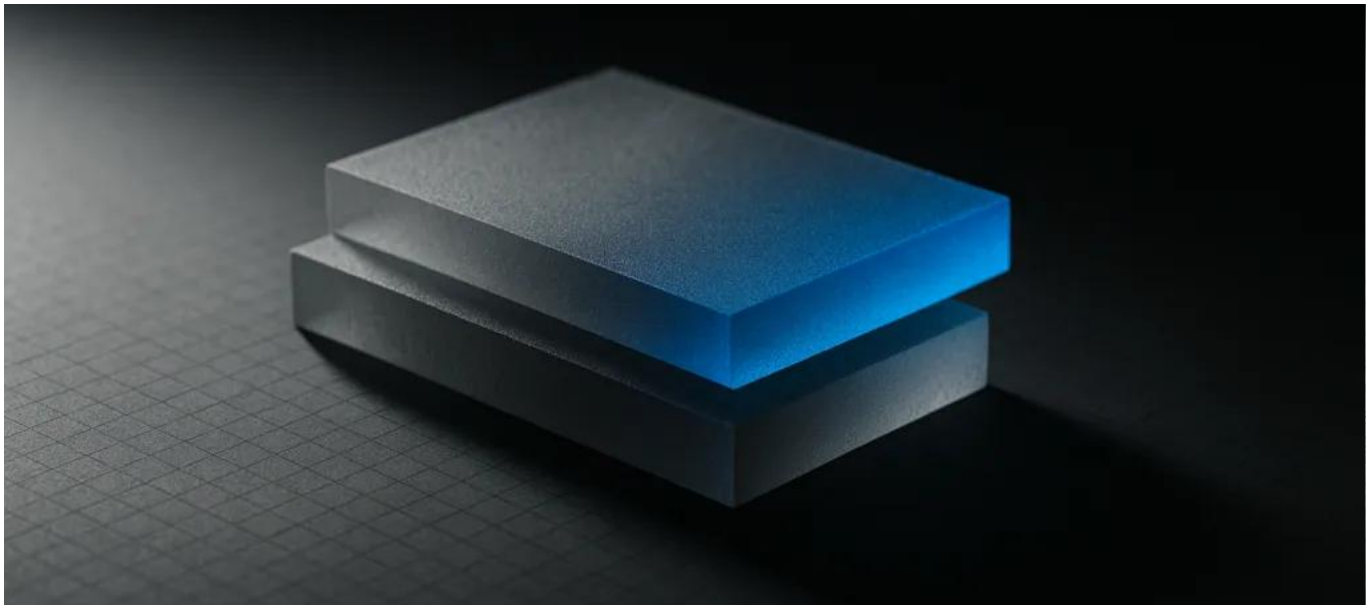

OpenAI vs Anthropic vs Gemini Prompt Caching: The Honest Cost Comparison for CTOs

Caching is a billing model choice now. The same prompt, the same tokens and the same traffic pattern produce three different invoices depending on whether your provider charges a write multiplier, a TTL-tiered write, or rent by the hour.

That last one is the part most teams miss. Google does not discount your reads at all: Gemini context caching is metered as storage, $0.40 per 1M tokens per hour under 200k tokens and $4.50 per 1M tokens per hour above, on top of standard token rates, per the Gemini API pricing docs. Your bill scales with wall-clock time. OpenAI and Anthropic both scale with requests.

Two September changes made this a live decision

Anthropic raised Claude Sonnet 5 input pricing from $2 to $3 per 1M tokens on September 1, with output going $10 to $15, confirmed by Anthropic. Because Anthropic derives every cache tier as a multiple of base input, that single change repriced the whole stack underneath it: 5-minute writes went $2.50 to $3.75/M, 1-hour writes $4 to $6/M, cache hits $0.20 to $0.30/M. Nobody changed a line of code and cached workloads got 50% more expensive.

Then on August 20, OpenAI shipped a Prompt Caching dashboard reporting cache hit rate, cache reads per write, and the split between cache-read, cache-write and uncached tokens. That matters because the economics of every caching model hinge on one ratio, reads per write, that most teams have been guessing at.

OpenAI charges 1.25x to write and 0.1x to read

OpenAI bills cached input at 0.1x uncached and cache writes at 1.25x uncached, per the prompt caching guide. On chat-latest that is $5.00/M uncached, $0.50/M cached, implying $6.25/M to write. The minimum cacheable prefix is 1,024 tokens, then 128-token increments, and TTL now runs up to 24 hours on GPT-5.5 and later, up from the old 5 minutes to 1 hour in-memory behaviour. One change worth flagging: cache writes used to be free in automatic mode. Flexera's breakdown notes that starting with GPT-5.6, both automatic and explicit modes bill writes at 1.25x.

Anthropic sells you a TTL dial, and Fable 5.1 is the outlier

Anthropic charges 1.25x base input for a 5-minute write, 2x for a 1-hour write, and 0.1x for a hit. On Opus 5 that is $5/M base, $6.25/M 5-min write, $10/M 1-hour write, $0.50/M hits. The interesting exception is Fable 5.1 and Mythos 5.1, where cache reads are $0.25/M against a $10/M input base. That is 0.025x, four times better than the standard 0.1x ratio, which I wrote about in more detail when Anthropic cut those read prices.

Gemini publishes no read discount at all, only storage rent

Gemini publishes no cache-write multiplier and no discounted per-request read rate. You pay standard token rates plus storage rent. The 200k token boundary is an 11.25x step rather than a gentle curve.

ATTRIBUTE	OPENAI	ANTHROPIC (SONNET 5)	GEMINI
Cache read	0.1x input ($0.50/M on chat-latest)	**0.1x input ($0.30/M)**	no discounted read rate
Cache write	**1.25x input ($6.25/M)**	1.25x (5-min) / 2x (1-hr) = $3.75 or $6.00/M	no write multiplier published
Storage charge	**none published**	**none published**	$0.40/M/hr under 200k; $4.50/M/hr above
Max TTL	**24 hours (GPT-5.5+)**	1 hour (paid tier)	billed hourly, no fixed tier
Minimum prefix	1,024 tokens, then 128-token steps	not published in this data	not published in this data
Hit-rate visibility	**Prompt Caching dashboard, Aug 20 2026**	not published in this data	not published in this data

Note the empty cells. I am not going to invent Anthropic's minimum prefix or Google's write behaviour to make the table look tidy. Where the research does not say, the table does not say.

The reads-per-write ratio decides everything

Under OpenAI and Anthropic, caching pays off the moment you read a cached prefix more than roughly three times inside its TTL. Write once at 1.25x, read at 0.1x: by the fourth request you are ahead of paying full price every time. Anthropic's 1-hour tier doubles the write cost, so it needs a fatter read count to justify itself, and it only makes sense when your gap between requests exceeds five minutes but stays inside the hour.

Gemini inverts the question. Reads get no discount, so the calculation is purely whether the

storage rent is cheaper than re-sending the tokens. For a 150k-token context held for an hour, you pay $0.06 in storage. Cross 200k and the same hour costs you 11.25x more per token. **That cliff is the thing to design around.**

MY READ

The OpenAI dashboard release is the more consequential of the two September events, even though the Anthropic price rise got the headlines. A 50% input increase is a number you can model on a spreadsheet. A cache hit rate you have never measured is an unknown multiplier on your entire inference bill, and most teams I talk to assume their hit rate is far higher than it is.

Pick by traffic shape, not by brand

Pick OpenAI if your prompts are long-lived and re-read across hours. The 24-hour TTL on GPT-5.5+ has no equivalent elsewhere, and the dashboard means you can verify the savings instead of assuming them. The 1,024-token minimum prefix rules out small system prompts, so this suits agents with heavy tool schemas and document context.

Pick Anthropic if you are on Fable 5.1 or Mythos 5.1 and your workload is read-heavy. A 0.025x read multiplier against a $10/M base is the cheapest cached read in this comparison in ratio terms, and it changes the break-even math enough that patterns which did not pay off before now do. On Sonnet 5 and Opus 5 the story is ordinary: standard 0.1x reads, a TTL dial you pay for, and a base price that just went up.

Pick Gemini if you hold a large context steady and query it rarely, or if your context is genuinely huge and re-transmitting it is the dominant cost. Storage pricing rewards low request volume against a big fixed prompt. It punishes chatty agents, because you pay rent whether or not anyone asked a question that hour.

Fixing your prefixes beats switching vendors

Before you move providers, fix your prefixes. DSPy 3.3.0's ReActV2 reports up to roughly 50% cost reduction on some workloads purely from better prompt-cache prefix reuse, with no change in model quality. That is the same order of magnitude as the entire Sonnet

5 price rise, available without renegotiating anything. Ordering your prompt so the stable parts sit in front and the volatile parts sit behind is unglamorous work that pays more than most vendor switches.

My take: for the typical team running a production agent with a fat system prompt and steady traffic, OpenAI is the default right now, mostly because of measurement. The 24-hour TTL and the 1.25x write are competitive without being decisive. **The dashboard is decisive**, because caching economics come down to a ratio, and OpenAI is the only one of the three that currently hands you that ratio. If you are already on Anthropic and running Fable 5.1 or Mythos 5.1, stay: the 0.025x read rate beats what either competitor offers, and moving would be a downgrade.

One thing I cannot tell you from the published pricing is how these three behave under real cache eviction pressure. Advertised TTL is a ceiling, not a guarantee, and I have not seen credible measurement of actual retention rates under load on any of the three. If you have production numbers on that, I would genuinely like to compare notes, so send them over and let us argue about them.

WRITTEN BY

Artur Markus

Software engineer in Basel building AI infrastructure and agentic systems, with a background in pharmaceutical automation, GxP compliance, and validation. I help teams turn AI from a chatbot into a reliable operational layer.

Book a meeting →