Claude Models Explained: Haiku vs Sonnet vs Opus (November 2026)

Edition of November 13, 2026 — facts verified November 13, 2026. Current edition always at https://tygartmedia.com/claude-models-comparison/.

Direct Answer (November 13, 2026): Claude models span four capability tiers (all with 1M context / 128K output): **Haiku 5.5 ($0.10/$0.50 MTok ≤100k tokens, $0.50/$2.50 >100k)** for fast high-volume routing, **Sonnet 5.5 ($2/$10 MTok)** for the best speed-and-intelligence balance across everyday production work, **Opus 5.5 ($4/$20 MTok)** for deep code refactoring and technical architecture, and **Fable 5.2 ($8/$40 MTok)** as the top capability flagship.

Updated November 13, 2026

Updated October 28, 2026: Made infographic diagrams clickable to open full-size images.

Updated November 2, 2026 — added direct answers on Claude model tiers, coding recommendations, agent architecture, and value tradeoffs, sourced from Anthropic's published model and pricing documentation.

Official links: Try the models (claude.ai) · Official model docs · API console

Comparison note: the tier-by-tier comparisons below remain valid. As of **November 13, 2026**, Anthropic's current lineup is **Claude Fable 5.2** (`claude-fable-5-2`, top tier above Opus; **$8 input / $40 output per MTok**; cache writes $10/$16, cache hits $0.20, batch $4/$20; 1M context / 128K output; prior Fable 5.1 at $10/$50 is legacy), **Claude Opus 5.5** (current Opus; $4/$20; launched September 22, 2026), **Claude Sonnet 5.5** (current Sonnet; released **September 28, 2026**; $2/$10 per MTok standard pricing; cache reads $0.10), and **Claude Haiku 5.5** (launched **October 7, 2026**; the new cheapest model — $0.10/$0.50 per MTok for prompts of 100k tokens or fewer, $0.50/$2.50 above). Opus 5 ($5/$25) and Sonnet 5 remain listed but are superseded. Opus 4.8 and Sonnet 4.6 are legacy; Sonnet 4.5 retires November 30, 2026. Full details: the Claude Fable 5 Complete Guide.

Last refreshed: November 13, 2026

Model Accuracy Note — Updated October 12, 2026

Lineup currency (Oct 2026, verified): Sonnet 5.5 ($2/$10; cache read $0.10), Opus 5.5 ($4/$20), Haiku 5.5 ($0.10/$0.50 for prompts ≤100k tokens), Fable 5.2 ($8/$40; legacy Fable 5.1 $10/$50). Superseded (still listed): Opus 5 ($5/$25), Sonnet 5 ($2/$10 — Anthropic has blanked its base price cells). Legacy: Opus 4.8 ($5/$25), Sonnet 4.6 ($3/$15); Sonnet 4.5 retires November 30, 2026. *Prior note (superseded):* Prior flagship claim (legacy): Fable 5.1 ($10/$50). Current flagship is **Claude Fable 5.2 ($8/$40)**, alongside Opus 5.5 ($4/$20), Sonnet 5.5 ($2/$10), and Haiku 5.5 ($0.10/$0.50 ≤100K). Where this article references Opus 4.6 or earlier models, those references are historical. See current model tracker →. See current model tracker →

Claude AI · Fitted Claude

Anthropic's model lineup is organized around three tiers — Haiku 5.5, Sonnet 5.5, and Opus 5.5 — each representing a different point on the speed-versus-intelligence spectrum. Understanding which model to use, and which API string to call it with, saves both time and money. This is the complete November 2026 reference.

Quick answer:

Haiku = fastest and cheapest, best for high-volume simple tasks. Sonnet = the balanced workhorse, right for most things. Opus = the heavyweight, use when quality is the only metric. For the API, always use the full model string — never just "claude-sonnet" without the version number.

Where Sonnet Wins

Where Sonnet wins.

Sonnet is not a compromise — it's the right tool for the majority of professional tasks. Writing, research, summarization, drafting, analysis, code generation, SEO work, email, strategy — Sonnet handles all of it at a level that's indistinguishable from Opus for most outputs. The difference shows up at the edges: highly ambiguous problems, tasks requiring multiple competing constraints to be held simultaneously, or situations where the consequences of a slightly wrong answer are significant.

For production API workloads, Sonnet's cost advantage is substantial. Running high-volume content or data pipelines on Opus instead of Sonnet multiplies costs without proportional quality gains on most tasks.

Where Opus Wins

Opus earns its premium on genuinely hard problems. Complex multi-step reasoning where the chain of logic matters. Legal or technical documents where precision at every sentence is required. Strategic analysis where you need the model to hold and weigh competing frameworks simultaneously. Code debugging on complex, unfamiliar systems where Sonnet gives you the obvious answer and Opus finds the non-obvious one.

I use Opus specifically for: client strategy documents where I'm synthesizing months of context, complex GCP architecture decisions, and any task where I've tried Sonnet and felt the output was a notch below what the problem deserved. That's a smaller subset of work than most people assume.

The Practical Routing Rule

The practical routing rule.

Use Sonnet when: the task is well-defined, the output type is familiar, and quality at the 90th percentile is sufficient. That's most professional work.

Use Opus when: the task is genuinely novel, involves high-stakes judgment, requires deep multi-step reasoning, or you've already run it on Sonnet and the output wasn't quite right.

Use Haiku when: you need the same operation at scale, latency matters more than depth, or cost is the primary constraint.

The Decision Framework

The decision framework.

Use Haiku when: same operation at high volume, output is constrained/structured, cost and speed matter, real-time latency required.

Use Sonnet when: any standard professional task — writing, coding, analysis, research. This should be your default 90% of the time.

Use Opus when: the task is genuinely hard, involves novel reasoning, Sonnet's output wasn't quite right, or quality is the only variable that matters regardless of cost.

For full pricing details, see Anthropic API Pricing. For a Haiku deep-dive, see Claude Haiku 4.5: Pricing, Use Cases, and API String. For the Opus vs Sonnet head-to-head, see Claude Opus 4.8 vs Sonnet.

The Three-Tier Model Architecture

Three seats. Version names change; the pyramid does not.

What is the difference between Claude Opus, Sonnet, and Haiku?

Anthropic's model lineup is organized as a three-tier hierarchy balanced across reasoning capability, processing latency, and token pricing:

- **Claude Haiku (Haiku 5.5):** The fastest and most cost-efficient tier. Listed at $0.10 / MTok input and $0.50 / MTok output for prompts up to 100K tokens ($0.50 / $2.50 above 100K; cache reads from $0.01 / MTok). Haiku is built for lightweight, latency-critical operations—such as high-volume classification, metadata extraction, deterministic routing, and customer support triage—where execution speed and low cost outweigh deep multi-step deduction.

- **Claude Sonnet (Sonnet 5.5):** The primary production workhorse. Listed at $2.00 / MTok input and $10.00 / MTok output (cache read $0.10 / MTok). Sonnet combines near-flagship intelligence with high throughput, making it the default recommendation for roughly 90% of professional workloads, including code generation and review, complex document synthesis, data extraction, and multi-turn agentic workflows.
- **Claude Opus (Opus 5.5):** The current Opus-tier model below Claude Fable 5.1. Listed at $4.00 / MTok input and $20.00 / MTok output (cache read $0.20 / MTok). Opus delivers maximum reasoning depth, nuanced contextual comprehension, and sustained accuracy on long-horizon logical chains. It serves as the daily driver for complex software architecture, novel algorithmic engineering, and high-stakes analytical tasks where precision at every sentence is critical.

At the apex of the lineup, Anthropic also offers **Claude Fable 5.1** ($10.00 input / $50.00 output per million tokens; cache read $0.25 / MTok), engineered specifically for autonomous, long-horizon agentic workflows that repeatedly re-read extensive repository contexts.

Is Opus 5.5 worth the extra cost over Sonnet 5?

Version note: In Anthropic's current published rate card, **Sonnet 5.5** is the active flagship Sonnet release ($2.00 input / $10.00 output per million tokens; cache read $0.10), while **Sonnet 5** remains listed on Anthropic's legacy rate card at the same base token rates ($2.00 input / $10.00 output; cache read $0.20). Current **Opus 5.5** lists at $4.00 / MTok input and $20.00 / MTok output (cache read $0.20), exactly **2× the cost** of the Sonnet line. (Legacy Opus 5 remains listed at $5.00 / $25.00 per million tokens.)

Whether Opus 5.5 is worth the 2× premium depends entirely on your failure-risk profile and task structure:

- **When Opus 5.5 delivers superior value:** On genuinely novel problems, high-stakes legal or financial synthesis, and complex multi-file architectural refactors where a subtle logical flaw invalidates the entire output. If an error in Sonnet requires an experienced engineer 45 minutes to debug and rewrite, paying an extra $2.00 input / $10.00 output per million tokens to get first-pass correctness from Opus 5.5 is overwhelmingly cost-positive.
- **When Sonnet delivers better value:** On standard, well-defined professional tasks—routine API endpoints, feature implementation against clear specs, test authoring, editorial drafting, and structured summarization. Sonnet 5.5 operates at 90th-percentile quality across these domains at half the token cost and noticeably faster latency, making a blanket upgrade to Opus 5.5 an unnecessary operational expense.

Which Claude model has the largest context window?

In Anthropic's current published lineup (Claude Fable 5.1, Claude Opus 5.5, Claude Sonnet 5.5, and Claude Haiku 5.5), every model features the identical maximum input context window: **1 million tokens (1M tokens)**, paired with a maximum generation limit of **128K tokens**. Legacy models (such as Haiku 4.5) were capped at a 200K token context window.

Because context capacity is uniform across the entire modern lineup, context size is no longer a selection constraint when choosing between Haiku, Sonnet, Opus, and Fable. Instead, model selection is determined strictly by reasoning depth, execution latency, and token economics. For deep-dive architectural analysis, token threshold benchmarks, and prompt structure guidelines, see our dedicated guide to the Claude context window.

Which Claude model should I use for coding?

For software development and code generation, the practical recommendation follows the hub's core decision framework:

- **Default to Claude Sonnet (Sonnet 5.5):** Sonnet should be your primary coding engine for roughly 90% of development tasks. It powers everyday code generation, bug fixing, test writing, PR reviews, and script authoring inside tools like Claude Code and IDE integrations. At $2.00 / MTok input and $10.00 / MTok output, it delivers the optimal balance of code accuracy and rapid interactive response.
- **Escalate to Claude Opus (Opus 5.5) for complex architecture:** Use Opus 5.5 ($4.00 / $20.00 per million tokens) when you encounter difficult architectural decisions, novel algorithms with unfamiliar edge cases, deep multi-file refactoring spanning complex dependency graphs, or when Sonnet's initial solution reveals logical hallucinations.
- **Route boilerplate and syntax linting to Haiku (Haiku 5.5):** For automated CI/CD linting, docstring generation, simple regex extraction, or parsing structured syntax trees at scale, route to Haiku ($0.10 / $0.50 per million tokens) to minimize pipeline costs.

Should I use Haiku for simple tasks to save money?

Yes—provided your workload stays strictly above the quality floor defined in the three-tier model architecture. Haiku 5.5 is priced from $0.10 per million input tokens and $0.50 per million output tokens (prompts ≤ 100K tokens), representing a 95% discount compared to Sonnet 5.5 and a 97.5% discount compared to Opus 5.5.

The routing heuristic:

- **Safe Haiku workloads:** Deterministic classification, keyword and entity extraction, sentiment scoring, routing user prompts to specialized tools, lightweight summaries of clean transcripts, and JSON schema formatting where the prompt provides clear constraints.
- **The quality floor:** Haiku should not be used when tasks require nuanced editorial tone, multi-step deductive reasoning, unconstrained problem solving, or synthesizing contradictory reference materials. If downstream human intervention or repeated re-prompts are needed to correct Haiku's output, the token savings evaporate immediately. For complete per-million token pricing and rate limit tiers, see our reference on Claude API pricing, token costs and rate limits.

What is the best Claude model for building AI agents?

For building autonomous and multi-turn AI agents, Tygart Media's published operator record and production fleet routing establish a multi-tiered pattern rather than a single-model choice:

- **Core agent reasoning and synthesis: Claude Sonnet (Sonnet 5.5).** As detailed in our production guide on Building Your First Agentic Workflow with Claude's Agent SDK, Sonnet serves as the primary driver for tool selection, planning, and final output synthesis. Its combination of structured tool calling fidelity and $2/$10 pricing prevents multi-turn agent loops from exhausting budgets.
- **High-frequency inner loop execution: Claude Haiku (Haiku 5.5).** For subagent routing, parameter extraction, and intermediate step evaluation, routing inner loops to Haiku significantly reduces token burn across high-turn workflows without degrading agent reliability.
- **Long-horizon autonomy and deep context loops: Claude Fable 5.2 and Opus 5.5.** When agents operate across multi-hour autonomous sessions or manage persistent cross-session memory stores (see Use Cases for Claude Managed Agents), Fable 5.2 ($8/$40 list; $0.20 cache read) and Opus 5.5 ($4/$20 list; $0.20 cache read) provide the reasoning endurance necessary to prevent agent drift. Always enforce strict `max_turns` ceilings and token budgets on production loops.

How do I choose between Claude's API models for a production app?

To choose between Claude's API models for a live production application, implement a three-step selection procedure grounded in operational failure risk:

1. **Audit task predictability and volume:** Map application endpoints into discrete latency and reasoning tiers. High-throughput, stateless endpoints with constrained schemas (intake parsing, tagging, search query re-writing) map directly to `claude-haiku-5-5`.
2. **Default the core application loop to Sonnet:** Set `claude-sonnet-5-5` as the baseline model across primary customer-facing features, multi-turn chat, data analysis, and code synthesis. Pin explicit, versioned model strings in your configuration rather than unversioned aliases.
3. **Reserve frontier reasoning for gated escalations:** Introduce `claude-opus-5-5` or `claude-fable-5-2` only at specific escalation gates—such as automated exception recovery, complex legal/financial generation, or high-stakes reasoning where an output error carries high business risk.

For comprehensive per-token rate cards, prompt caching multipliers, and batch API volume discounts, consult our complete breakdown of Claude API pricing, token costs and rate limits.

Full Claude model lineup — November 2026

Model	Tier	Best for	Input $/MTok	Output $/MTok	Context
Claude Fable 5.2	Current flagship	Most demanding reasoning & agentic work (128K output)	$8	$40	1M tokens
Claude Fable 5.1	Legacy	Superseded by Fable 5.2; prior $10/$50 rate	$10	$50	1M tokens
Claude Opus 5.5	High capability	Complex reasoning, long-horizon agentic coding	$4	$20	1M tokens
Claude Opus 5	Legacy	Superseded by Opus 5.5; still listed	$5	$25	1M tokens
Claude Sonnet 5.5	Balanced	Production apps — best speed/intelligence ratio	$2	$10	1M tokens
Claude Sonnet 5	Superseded	Anthropic has blanked its base price cells; still listed	$2	$10	1M tokens
Claude Haiku 5.5	Fast/efficient	High-volume, latency-sensitive, cost-sensitive	$0.10*	$0.50*	1M tokens
Claude Haiku 4.5	Legacy	Superseded by Haiku 5.5; still listed	$1	$5	200k tokens

Pricing from platform.claude.com as of October 12, 2026. Claude Fable 5 launched June 9, 2026 as the new most capable widely-released model. Claude Haiku 5.5 launched October 7, 2026 as the new cheapest model ($0.10/$0.50 per MTok for prompts of 100k tokens or fewer; $0.50/$2.50 above). Fable 5.2 lowers the flagship API rate to $8/$40 per MTok (legacy Fable 5.1 was $10/$50; Mythos previews were restricted/retired from standard comparison).*

Claude vs competitors — June 2026

Compare shapes first. Then check live rates on each vendor.

Platform	Flagship model	Key strength	Input $/MTok
Anthropic	Claude Fable 5.2	Reasoning, agentic coding, 1M context / 128K output	$8
OpenAI	GPT-5.5	Agentic tasks, coding, cross-tool workflows	Contact OpenAI
Google	Gemini 3.5 Flash (GA June 9) / Gemini 2.5 Pro (stable)	Multimodal, Google ecosystem integration	See ai.google

Competitor data sourced from openai.com and deepmind.google/models/gemini as of June 9, 2026.

Anthropic structures its models around a consistent naming pattern: a Greek letter indicating capability tier (Haiku → Sonnet → Opus, low to high) and a version number indicating the generation. The current generation is the 5.x series.

Model	API String	Context Window	Best for
Claude Haiku 5.5	claude-haiku-5-5	1M tokens	Classification, tagging, high-volume pipelines
Claude Sonnet 5.5	claude-sonnet-5-5	1M tokens	Most production work, writing, analysis, coding
Claude Opus 5.5	claude-opus-5-5	1M tokens	Complex reasoning, research, quality-critical
Claude Fable 5.2	claude-fable-5-2	1M tokens	Top-tier flagship reasoning & autonomous workflows

Claude Haiku 5.5: Speed and Cost Efficiency

Haiku is Anthropic's fastest and least expensive model. It's built for tasks where throughput and cost matter more than maximum reasoning depth — think classification pipelines, metadata generation, content tagging, simple Q&A at volume, or any workload where you're making thousands of API calls and can't afford Sonnet pricing at scale.

Haiku 5.5 is priced by prompt length: $0.10 input / $0.50 output per MTok for prompts of 100,000 tokens or fewer, rising to $0.50 / $2.50 above that — about 90% cheaper per token than Haiku 4.5 at short prompts.

Don't mistake "cheapest" for "bad." Haiku handles everyday language tasks competently. What it can't do as well as Sonnet or Opus is maintain coherence across very long context, handle subtle nuance in complex instructions, or produce writing

that reads like a human crafted it. For structured outputs and clear-cut tasks, it's excellent.

When to use Haiku: batch content generation, automated tagging and classification, chatbot applications where responses are short and structured, high-volume data processing, anywhere you're cost-sensitive at scale.

Legacy — Claude Sonnet 4.6: The Previous Production Workhorse

Sonnet is the model most developers and knowledge workers should default to. It sits at the sweet spot of the capability-cost curve — significantly more capable than Haiku at complex tasks, significantly cheaper than Opus, and fast enough for interactive use cases.

Sonnet handles long-document analysis well, produces writing that requires minimal editing, follows complex multi-part instructions without drift, and codes competently across most languages and frameworks. For the overwhelming majority of real-world tasks, Sonnet is the right choice.

When to use Sonnet: article writing, code generation and review, document analysis, customer-facing AI features, research summarization, agentic workflows that need a balance of quality and cost.

Legacy — Claude Opus 4.8: Maximum Capability

At its launch, Opus 4.8 was Anthropic's most powerful model — and its most expensive. It's built for tasks where you need maximum reasoning depth: complex strategic analysis, intricate multi-step problem solving, long-horizon planning, nuanced evaluation work, or any scenario where you'd rather pay more per call than accept a lower-quality output.

Opus is not the right default. The cost premium is real and meaningful at scale. The right question to ask before routing to Opus is: "Will a human reviewer actually tell the difference between Sonnet and Opus output on this task?" If the answer is no, use Sonnet.

When to use Opus: high-stakes strategic documents, complex legal or financial analysis, research that requires synthesizing across many sources with genuine insight, tasks where the output gets published or presented to executives without further editing.

Legacy — Claude Opus 4.8 vs Sonnet: The Practical Decision

Ask what fails if the answer is wrong. That picks the seat.

Task Type	Use Sonnet	Use Opus
Article writing	■ Usually	Long-form flagship only
Code generation	■ Most tasks	Complex architecture
Document analysis	■ Standard docs	High-stakes, nuanced
Strategic planning	Good enough	■ When stakes are high
High-volume pipelines	■ Or Haiku	■ Too expensive
Interactive chat	■ Best fit	Overkill for most

Claude Sonnet 5.5: The Current Sonnet

Anthropic follows a consistent release cadence — major model generations are announced publicly and the naming convention stays stable. The current top-tier flagship is Claude Fable 5.2 ($8/$40 MTok). Claude Sonnet 5 shipped June 30, 2026 and held the production default until Claude Sonnet 5.5 arrived September 28, 2026; Claude Opus 5 shipped July 24, 2026, replacing Opus 4.8 *(legacy — still listed)*, and Claude Opus 5.5 (launched September 22, 2026) is now the current Opus. As of November 13, 2026, the current models are Claude Fable 5.2 (top tier flagship), Claude Opus 5.5, Claude Sonnet 5.5, and Claude Haiku 5.5 (launched October 7, 2026 — the new cheapest model). Sonnet 5 is superseded (Anthropic has blanked its base price cells). Sonnet 4.6, Opus 4.7, and Opus 4.6 are legacy versions and should not be used for new integrations; Sonnet 4.5 retires November 30, 2026.

When new models release, Anthropic typically maintains the previous generation in the API for a transition period. Production applications should always pin to a specific model version string rather than using a generic alias, so new model releases don't silently change your application's behavior.

How to Use Model Names in the API

Always use the full versioned model string in API calls. Generic strings like `claude-sonnet` without a version may resolve to different models over time as Anthropic updates defaults.

```
# Current production model strings (November 2026)
claude-fable-5-2   # Current flagship ($8/$40 MTok)
claude-opus-5-5    # Opus tier ($4/$20 MTok)
claude-sonnet-5-5  # Balanced default ($2/$10 MTok)
claude-haiku-5-5   # Fast, cheap ($0.10/$0.50 <=100K)
# Always pin exact versioned model IDs from platform.claude.com/docs
```

Frequently Asked Questions

What is the best Claude model?

Claude Fable 5.2 ($8/$40 MTok) is Anthropic's most capable flagship model, while Claude Sonnet 5.5 ($2/$10 MTok) is the best choice for most everyday production use cases — it offers the best balance of capability, speed, and cost. Use Opus only when the task genuinely requires maximum reasoning depth. Use Haiku for high-volume, cost-sensitive workloads.

What is the difference between Claude Sonnet 5 and Claude Opus 5?

Note: Sonnet 5 and Opus 5 are superseded but remain listed; Sonnet 5.5 and Opus 5.5 are current. Sonnet is the balanced mid-tier model — faster, cheaper, and suitable for most production tasks. Claude Fable 5.2 is the most capable current model; Opus is the tier below it, significantly more expensive, and best reserved for complex reasoning tasks where quality is the primary consideration. For most writing, coding, and analysis tasks, Sonnet's output is indistinguishable from Opus at a fraction of the cost.

What are the current Claude model API strings?

As of October 2026: claude-haiku-5-5 (Haiku 5.5); for Sonnet 5.5 and Opus 5.5, pin the full versioned strings published at platform.claude.com/docs. Always use the full versioned string in production code to avoid silent behavior changes when Anthropic updates model defaults.

Is Claude Sonnet 5.5 available?

Yes. Claude Sonnet 5.5 is the current production-default Sonnet (released September 28, 2026), replacing Sonnet 5. It runs at $2 input / $10 output per MTok, with cache reads at $0.10 per MTok (halved from $0.20 on October 7, 2026). Sonnet 5 remains listed but is superseded — Anthropic has blanked its base price cells. The current top tier is Claude Fable 5.2 ($8/$40), with Claude Opus 5.5 ($4/$20) as the current Opus.

The Next Step · Research Verdict

Which Claude path makes sense for you?

Default to **Claude Sonnet (Sonnet 5.5)** for ~90% of development, agentic, and professional tasks — it delivers near-flagship intelligence with rapid throughput. Escalate to **Claude Opus (Opus 5.5)** or Fable for extreme architectural depth, and route repetitive bulk tasks to **Haiku 5.5**.

Try Sonnet & Opus at claude.ai →

Read official Model docs at Anthropic →

Check API token pricing at Anthropic Console →

Compare OpenAI Models & Pricing →

Primary & Reference Links Cited on This Page

- Try the models (claude.ai) — `https://claude.ai`

- Official model docs — `https://platform.claude.com/docs`
- API console — `https://platform.claude.com`
- Check API token pricing at Anthropic Console → — `https://platform.claude.com/`
- Compare OpenAI Models & Pricing → — `https://openai.com/chatgpt/pricing/`
- Tygart Media: the Claude Fable 5 Complete Guide
- Tygart Media: current model tracker
- Tygart Media: https://tygartmedia.com/wp-content/uploads/2026/08/claude-invisible-agent-layer.jpg
- Tygart Media: https://tygartmedia.com/wp-content/uploads/2026/08/anthropic-api-key-before-you-start.jpg
- Tygart Media: Anthropic API Pricing
- Tygart Media: Claude Haiku 4.5: Pricing, Use Cases, and API String
- Tygart Media: Claude Opus 4.8 vs Sonnet
- Tygart Media: https://tygartmedia.com/wp-content/uploads/2026/08/claude-three-tier-architecture.jpg
- Tygart Media: Claude context window
- Tygart Media: Claude API pricing, token costs and rate limits
- Tygart Media: Building Your First Agentic Workflow with Claude's Agent SDK
- Tygart Media: Use Cases for Claude Managed Agents
- Tygart Media: https://tygartmedia.com/wp-content/uploads/2026/08/claude-vs-competitors-cards.jpg
- Tygart Media: https://tygartmedia.com/wp-content/uploads/2026/08/claude-opus-vs-sonnet-fork.jpg