

The Cloud Agent Factory — v3

Audits, sites, and SEO as a production agent fleet — plus accounts, Managed Agents 101, the conference playbook, and live mission control

For: Dennis Yu · Dylan Haugen · Cam Hazzard Execution: Josh Healy Date: June 12, 2026 (v3)

1. Executive summary

Today every deep audit runs interactively in Cowork on Dennis's Max account at ~\$20 each in extra usage. That works at 25/day; it does not work at 50–150/day conference batches, and it makes Dennis the bottleneck. The fix is not more personal accounts and not abandoning the ones we have — it's two lanes:

People keep their personal Max 20x accounts — the richest included-usage deal Anthropic sells — for prototyping, QA, and judgment work in Cowork.

Machines run in one company Claude Console org (API) — where Claude Managed Agents runs our existing skills as cloud agents: webhook-triggered, cron-scheduled, massively parallel, pay-per-token.

\$20 → \$4.50 Deep-audit COGS, phased	~\$470 Full 150-person conference, Phase 1	~30 sec To launch 150 parallel sessions	\$0 Servers and DevOps hires
---	---	---	--

Immediate forcing function: the ~150-name Digital Wichita list arrives within 24 hours; the event is in ~7 days. Section 8 is the day-by-day runbook — including the live **mission-control big screen**: attendees watch their agent work on their phones while Dennis's screen shows the whole fleet (working demo files ship with this doc). v3 also adds the **Agent Registry** (§11): the fix for the clunky task dashboard and the documentation backbone for every agent we launch.

Metric	Today (Cowork)	Phase 1: Managed Agents	Phase 2: Pipeline + Batch
Deep audit COGS	~\$20	~\$15	~\$4.50
Quick audit COGS	manual	\$0.95 direct / \$0.50 batch	~\$0.30
150-person conference, all-in	impossible	~\$470	~\$170
50 audits/day — monthly tokens	~\$30,000 (naive)	~\$7,100	~\$2,400

2. Accounts: what we keep, what we add

Three different things get called "a Claude account." We need two of them, and we already own one.

	What it is	What it's for	Cost
Personal Max 20x HAVE — keep	Consumer subscription; ~20x Pro included usage + Cowork/Claude Code	Humans working interactively: prototyping skills, QA, client calls, one-off builds	\$200/mo each
Claude Console org ADD NOW — Dylan	Company API account: workspaces, keys, vaults, spend caps. Where Managed Agents + Batch run	The production fleet: webhook audits, conference batches, cron refreshes, site builds	Pay-per-token; no seats
Team / Enterprise org DEFER	Seat-based business chat plan. Premium seat ≈ 6.25x Pro per session, incl. Cowork + Code	Builder cohorts who don't own personal plans; central billing + admin controls	\$25–125/seat/mo

Answers, directly

"Do we keep personal accounts but also have a company account?" Yes — exactly that. Personal Max for people, Console org for machines. They bill separately, and that's a feature: seat money = tooling for humans; API money = COGS, reported per product line by workspace.

"Isn't the API less cost-efficient than our 20x plans?" For a human at a keyboard, yes — Max 20x included usage is subsidized and unbeatable, which is why we keep it. But plans subsidize *human hours*, not *unattended fleets*: a personal plan hits session limits, can't run 150 parallel sessions, can't take webhooks, and routing production through consumer accounts is what Anthropic's terms prohibit. Meanwhile our *marginal* cost today already IS API rates — that's what extra usage is. The fleet's \$4.50 audit beats the \$20 we pay right now, 4-to-1.

"The top business plan is only ~12x, not 20x — why downgrade?" Nobody downgrades. Team Premium is ~6.25x Pro per session — less than Max 20x — which is precisely why principals keep personal Max and we don't buy Team seats now. Team becomes relevant when AI Builder cohorts need company-paid seats. (v1 said buy Team in week 1 — corrected.)

The hand-off pattern: prototype a skill in Cowork on your personal account → drop the .md in the shared skills folder → Dylan/Josh attach it to the agent in Console. The personal lane invents; the company lane runs.

Ahrefs — checked live, June 12

We're on **Ahrefs Lite, billed monthly: 100,000 API units/mo, 61,397 used, resets June 19**. So yes, we have an Ahrefs API account — and it's now a tracked COGS line and a real constraint. Quick audits draw ~30–60 units each (150 people ≈ 5–9K units — fits); deep audits draw several hundred. At 1,500 audits/mo Lite will not hold: budget an upgrade or unit top-ups when volume arrives, and have the fleet log units-per-audit so the decision is made with data. Lucky break: the reset lands the week of Digital Wichita.

Funding the meters — exactly what to buy

Anthropic (tokens): there is no plan to upgrade on the API side — you just buy credits. Dylan, in the new Console org: **Settings** → **Billing** → **buy ~\$500 of credits today**, turn on auto-reload (e.g., +\$100 when the balance drops below \$50), and set the per-workspace spend caps from \$9. The usage tier (rate limits) rises automatically with cumulative credit purchases — \$500 plus Wichita-week spend clears the early tiers without talking to sales. Nothing changes on anyone's personal Max plan.

Ahrefs (units): Lite already includes the 100K units/mo we observed — marginal Ahrefs cost per audit is \$0 until we blow the allocation. When we do: buy pay-as-you-go unit top-ups in the Ahrefs dashboard (unused top-ups expire after ~3 billing months — buy as needed, don't stockpile) or upgrade the plan tier for a bigger allocation. Decide AFTER Wichita with logged units-per-audit in hand — one week of data turns the question into arithmetic.

3. Managed Agents 101 — for Josh, Dylan, and Cam

The 60-second mental model: Managed Agents has four nouns.

Noun	Plain English	Ours
Agent	The job description: model + system prompt + skills + tools + MCP connections. Created once in Console, referenced by ID	Spotlight Auditor · Site Builder · SEO Tuner · Content Factory
Environment	The workbench: an Anthropic-hosted Linux sandbox with bash, files, browsing	Anthropic cloud sandbox — we host nothing
Session	One shift: an agent instance working one task, with its own filesystem and history; resumable; deletable	One session per audit or site
Event	The log: messages in, results out, status updates — streamed and stored	What the QA queue and webhooks read

One audit, start to finish: webhook (or button) starts a session with {name, company, email, audit_type} → the agent loads skills, pulls Ahrefs/GSC via MCP (keys live in a **vault** — the agent never sees the real key, so prompt injection can't leak it) → researches, scores, writes, renders the PDF in its sandbox → completion webhook hands the PDF to WordPress/Keap → the session gets deleted. Wall clock: **~20–40 min** deep, **~3–6 min** quick.

What Josh actually clicks (the whole job): Console → workspaces with spend caps → Agents → New → paste system prompt → attach skill zips → add MCP URLs + vault credentials → copy the agent ID into the webhook-bridge config. That's the "DevOps." No servers, deploys, or containers exist anywhere in this plan.

Parallelism — "can we do them all at once?" Yes.

The API accepts **300 session-creates per minute** — a 150-person batch launches in ~30 seconds. Every session runs in its own sandbox, fully independent: 1 audit and 150 audits are the same architecture.

The governor is the org's **token-per-minute tier**, which rises with cumulative credit purchases. **Pre-fund the Console org the day it's created** (a few hundred dollars) so Wichita week isn't throttled at entry tier, and run batches in waves (30 → 60 → 60) — the API has acceleration limits that dislike cold spikes. Console → Settings → Limits shows the tier.

Caching is the throughput weapon: **cached input tokens don't count against the rate limit**, and all 150 audits share the same cached skill/rubric prompt. Fan-out gets cheaper AND faster per audit.

Steady state at 50/day is ~10–15 concurrent sessions — nowhere near any limit. Conference bursts are the only spike, and they're wave-able.

Cost anatomy of one Phase-1 deep audit: ~150K fresh input (\$1.50) + ~100K cache writes (\$1.25) + ~600K cached reads (\$0.60) + ~220K output incl. thinking (\$11.00) + ~45 min runtime (\$0.06) + ~25 searches (\$0.25) ≈ **\$14.66**. Output dominates — which is why Phase 2 moves synthesis to the Batch API at half price and hands data-gathering to cheaper models.

4. Architecture

CLIENT <i>never sees Claude</i>	STOREFRONT <i>we own — already live</i>	AGENT FLEET <i>Anthropic cloud</i>	BUILDERS <i>personal Max 20x</i>
Pays \$149 deep audit, \$99/mo Spotlight, or scans the conference QR	WP order form (Keap/Stripe) fires a webhook →	Webhook starts an Auditor session: skills + Ahrefs MCP + WP MCP + vaults	QA task lands in queue; builder reviews in Cowork, approves or steers
Receives PDF + live pages + 90-day plan	← BlitzAdmin REST publishes; Keap emails the PDF	← Completion webhook returns PDF + data; session deleted after delivery	Console dashboards: spend by workspace, session logs, weekly COGS report

Agents: Spotlight Auditor (strategist + mentions-harvester + gap-analyzer + KP-entity-SEO + AI-search-visibility, with a **person overlay** and a **business overlay** chosen at triage) · Site Builder (website-agent + TT25 + BlitzAdmin REST) · SEO Tuner (gap-analyzer + definitive-article-writer) · Content Factory (dollar-a-day + content-factory + definitive-article-writer, on cron). All carry boil-the-ocean + recursive-self-improvement-qa so quality compounds.

The glue — the only code anywhere: one webhook bridge (~200 lines, Cloudflare Worker or WP plugin): form → start session; completion → publish/email; nightly → delete delivered sessions. Claude Code writes and maintains it. Zero-code fallback: Zapier/Make ~\$50/mo. **Explicitly not built:** servers, queues, containers, agent loops, client logins, custom dashboards.

5. True cost per deliverable

Rates (Anthropic, June 2026): Fable 5 \$10/\$50 per MTok, cache hits \$1, Batch 50% off; Haiku 4.5 batch \$0.50/\$2.50; Sonnet 4.6 \$3/\$15 (batch half); web search \$10/1k; runtime \$0.08/hr. NEW: Ahrefs units are a second metered COGS (Lite = 100K/mo).

Deliverable	Phase 1 COGS	Phase 2 COGS	Ahrefs units	Notes
Roster triage (per name)	~\$0.07	~\$0.05	0–10	Haiku batch: identify, classify, route
Quick Audit	\$0.95 direct / \$0.50 batch	~\$0.30	30–60	Sonnet; direct = 3–6 min, for live demos
Deep Audit	~\$15	~\$4.50	200–500	Fable 5, max thinking on synthesis
Spotlight site build	~\$24	~\$15	~50	Per the Julian David recipe
Spotlight monthly refresh	~\$2	~\$1.70	~20	2 articles + monitoring
Vertical bundle monthly	~\$10	~\$7	~100	4 articles + DAD checks + maps monitor

Phase 1 = drop-in managed agent (week 1): caching + no idle re-reads takes \$20 → ~\$15 and unlocks parallelism.

Phase 2 = staged pipeline (weeks 5–8): deterministic collectors → Haiku batch extraction (~\$0.45) → ONE Fable 5 Batch synthesis call at max thinking (~\$3.50) → PDF render. Batch returns within 24h (usually much faster); rush jobs take the interactive path at ~\$15 (+\$50 rush fee).

6. Client pricing ladder

Clients buy outcomes — never tokens, seats, or logins. Free diagnosis → paid deep diagnosis → subscription.

Product	Price	COGS (Ph1 → Ph2)	Margin	Role
Quick Audit	Free	\$0.95 → \$0.30	lead magnet	Live today; now automated
Deep Audit — person OR business overlay	\$149 one-time (\$99 event code)	\$15 → \$4.50	90→97%	Credits toward packages
Spotlight — personal brand site	\$99/mo + \$149 setup	~\$8/mo all-in	~90%	The volume product
Knowledge Panel Package	\$499 one-time	\$20–35	~93%	Upsell from Deep Audit
Vertical Authority Bundle	\$299–499/mo	~\$10/mo	~96%	Roofers, landscapers, HVAC...
Youth / Athlete Spotlight	\$39/mo (sold to parents)	~\$4/mo	~90%	Julian David model
Maps Visibility System	\$1,500/loc/mo (unchanged)	\$60–100/mo	n/a	Agents raise margin; price holds

Floor logic: \$149 carries a 10x cushion even at Phase-1 COGS; the value anchor (competitors charge \$1,500+) matters more than our cost. **Stage-flash option:** a first-25-buyers flash at \$29 clears Phase-1 COGS (~\$15 tokens; Ahrefs units already in allocation) with a thin-but-real margin, creates scarcity, and reverts to WICHITA99 (\$99) when the 25 sell out. \$29 is the absolute floor — always a counted, expiring code so the \$149 anchor stays intact.

7. Vertical bundles: one pantry, many menus

Bundle = agent recipe + vertical overlay + price + landing page. Same agents, different overlay file (vocabulary, worked examples, scoring weights) — the Pure Green pack pattern. A new vertical is a one-day clone task.

Bundle	Recipe	Overlay	Price
Roofer / Landscaper / HVAC Authority	Auditor + Content Factory + SEO Tuner	Trade-specific hooks, certs, seasonal angles	\$299–399/mo
Founder Spotlight	Auditor + Site Builder + KP	Entity home + Knowledge Panel	\$99/mo
Marketing Manager (NEW — the Wichita persona)	Auditor (business overlay) + SEO Tuner	Channel performance, competitor gaps, AI-readiness — plus a 1-page exec summary addressed to their boss	\$149 audit → \$299/mo
Youth Athlete	Site Builder lite + harvester	Recruiting/NIL, parent-facing	\$39/mo
Agency white-label	Auditor at volume	Partner's brand on the PDF	\$59/audit, 25+

The marketing-manager overlay matters: they aren't the owner, so the win is making *them* look smart internally. The audit's last page becomes their pitch deck to the boss — that's what converts a \$0 attendee into a \$299/mo company.

8. Conference mode: batches of 50–150, and live on stage

Two modes, same fleet. The mistake to avoid: \$15-deep-auditing 150 unvetted names (\$2,250+) when a third may not own a business.

Mode 1 — Pre-event batch (the default)

Stage	What happens	Cost for 150
0 • Triage — all names	Haiku batch resolves identity from name + company (+ LinkedIn when provided), classifies — owner / marketing manager / personal brand / no business yet — and routes to the right overlay. Ambiguous identities flagged for human eyes, not guessed	~\$10, overnight
1 • Quick Audit — everyone	Sonnet batch, personalized 5-minute scorecard per person or company; staged in Keap, released at registration. "Your audit is waiting in your inbox" is the session opener	~\$75
2 • Deep audits — surgically	Comp the top ~15–20% from triage (real owners, hot signals); everyone else gets the \$99 event code	25 comped ≈ \$375 Ph1 / ~\$115 Ph2

Whole-conference total at 150 heads: **~\$470 Phase 1, ~\$170 Phase 2** — for 150 warm leads, inbox engagement before you speak, and the upsell baked into the talk.

Mode 2 — Live on stage ("run it while we talk")

QR on the opening slide → 30-second form → webhook fans out **direct** (non-batch) quick audits → results land by email in **3–6 minutes**, while you're still presenting. ~\$0.95/person; 100 walk-ins ≈ \$95. The 300-creates/minute limit means a ballroom launches in one minute; run waves of ~50 to respect acceleration limits.

Deep audits are NOT the live demo — 20–40 min is an awkward stage wait. The line: "*Your quick audit is in your inbox right now. Buy the deep audit with code WICHITA99 and it's on your breakfast table tomorrow*" — overnight batch, a second email touchpoint, and urgency. VIP exception: launch 3–5 interactive deep audits at minute zero and use one as the closing reveal (~\$15 each).

Registration data: use what registrants gave the organizer (name, company, email) plus public business data; deliver to the registered email with an unsubscribe.

The live experience — mission control on the big screen

Dennis's ask — attendees hit a URL, watch their agent work on their own phone, while the big screen shows the whole fleet like air traffic control — **is buildable, and the demo already exists**. The Managed Agents session event stream feeds three small endpoints on the webhook bridge:

Endpoint	Who uses it	What it shows
POST /api/start	The QR form	Starts the session, returns a session ID
GET /api/status/:id — 3-sec poll	Each attendee's phone	5-step live tracker: queued → researching → scoring → writing → delivered, with real agent events streaming underneath (Pulling Ahrefs... found 47 reviews, 4.8★) and the WICHITA99 upsell at the finish
GET /api/wall — 3-sec poll	Dennis's big screen	The fleet: a tile per attendee (first name + company + live phase), counters (started / running / delivered / avg time), live feed, arrivals-per-minute chart, running token-spend ticker

Two working demo files ship with this doc — open them in any browser: **wichita-mission-control.html** (big screen; simulated sessions arriving in waves) and **wichita-attendee-view.html** (phone view; 60-second autoplay). Self-contained, DEMO-badged, with the go-live JSON contract documented in each file header — they ARE the spec Claude Code wires to the real bridge June 15–18 (~half a day). Robustness choices: **polling, not websockets** (survives venue wifi); **first name + company only** on the big screen; and if anything dies mid-demo, every audit still delivers by email — the wall is theater, the email is the product.

Digital Wichita runbook (list ~June 13; event ~June 19)

Day	Action	Owner
Jun 12 (today)	Dylan creates the Console org, pre-funds past entry tier, makes the Audits workspace + spend cap, vaults the Ahrefs key. Claude Code scaffolds triage + quick-audit jobs	Dylan + Claude Code
Jun 13	List arrives → normalize to CSV (name, company, email, LinkedIn?)	Josh
Jun 14	Triage batch overnight (~\$10) → segmented roster + overlay per person	fleet
Jun 15–16	Quick-audit batch, all 150 (~\$75). Builders QA-sample 20; Dennis spot-checks 5. Claude Code wires mission-control wall + attendee live view to the bridge (demos shipped 6/12)	fleet + builders + Claude Code
Jun 17	PDFs staged in Keap; pick ~25 deep-audit comps from triage scores	Dylan + Cam
Jun 18	Dry-run the FULL live flow: QR → form → attendee tracker → wall on the projector → email lands. Launch comped deep audits (~\$375)	Josh
Jun 19	EVENT: QR live demo for walk-ins; WICHITA99 pitch; overnight batch for buyers	Dennis on stage
Jun 20	Deliver purchased deep audits "at breakfast"; Keap upsell to Spotlight / \$299 bundles	fleet + Dylan

Ahrefs check: quick audits (~5–9K units) fit the ~38K units left this cycle; the meter resets June 19, so deep-audit volume lands on a fresh 100K. Log units-per-audit during this run — it sets the upgrade math. **Wichita IS the Days 0–14 proof from v1**, pulled forward with a paying-audience deadline.

9. Access, logins, and governance

People (interactive): principals keep personal Max 20x — richest included usage, already paid, already loaded with our memory and connectors. Builders prototype skills there; production never routes through personal accounts. Team seats: deferred until AIBP cohorts need company-paid access (\$25 Standard / \$125 Premium with admin controls); certification still gates fleet permissions either way.

Machines (production) — the Console org Dylan is setting up:

Workspaces per product line (Audits / Sites / Content), each with its own API key and monthly spend cap. The per-workspace usage report IS the COGS report. Starting caps: Audits \$1.5K/mo, others \$500.

Vaults hold Ahrefs / WordPress / Keap credentials; agents and builders never see raw keys.

Certification = permission policy: Trainee → quick audits into a review queue · Certified Builder → deep audits + publish to client sites · Staff → DNS, Keap, billing-adjacent tools. College-partner certs (DSDT et al.) enter at the Trainee→Certified boundary.

Pre-fund on day one to clear entry-tier rate limits before Wichita; watch tier + cache hit rate in Console.

Clients: never get Claude anything — WordPress account on localservicespotlight.com for deliverables, Keap for billing. **Hygiene:** sessions deleted after delivery (not ZDR-eligible; fine for public marketing data — delete anyway). Human review before anything publishes under a client's name; PDFs carry "AI-assisted, reviewed by [builder]" per Anthropic's disclosure rules — already our certification model. Youth vertical contracts through parents; minors never touch Claude.

10. People: who does what

Person	Role
Dennis	Quality bar. His Cowork audits are the eval set every agent version must tie before promotion. On stage at Wichita
Dylan	Console org owner (in progress — correct instinct, right account type per §2). Storefront, Keap flows, bundle pages, pricing codes
Cam	Certification rubric, QA sampling quotas (100% → 1-in-5 → 1-in-10), Trainee→Certified promotions
Josh	Agent Operations, not DevOps — §3 is his onboarding doc. Console config, vault rotation, bridge health, session-log triage, weekly COGS report (tokens + Ahrefs units). Claude Code writes all code he "maintains"
Builders	QA queue, judgment calls, steering wayward sessions — on their own Max accounts

DevOps hire: still no. Nothing in this plan has a server. The triggers that would change the answer (self-hosted sandboxes for compliance, enterprise integration ownership, a real-time client dashboard product) remain unmet; a fractional contractor at 1–2 days/month covers any of them before a hire would.

11. Agent Registry: document the fleet, fix the dashboard

We're about to launch a lot of agents into this marketplace, and we already feel the pain Dennis named: the Task Library has 239 documented tasks, but the dashboard on top is clunky because the underlying docs are uneven. Fix the root cause, not the dashboard: **one registry, generated from the skill files themselves.**

The standard: every agent and task is one skill.md with required frontmatter — name, purpose, inputs, outputs, owner, version, cogs_per_run, ahrefs_units, eval_score, last_run, bundles[]. The 239-task library is the seed corpus; missing fields get filled by an agent pass, flagged for human confirm.

The registry: a script (Claude Code writes it) walks the skill folders and emits registry.json — the single source of truth for what exists, what it costs, who owns it, and which bundles sell it.

The dashboard: regenerated FROM the registry — filterable by bundle, vertical, owner, cost, eval score. The clunky dashboard gets replaced, not patched; as a pure rendering of the registry it can never drift from reality again.

Self-maintaining: every agent carries recursive-self-improvement-qa — each production run appends its actual cost, duration, and eval result to its own registry entry. Documentation that updates itself every time the fleet works.

Two truths, deliberately: the Console is runtime truth (agents, spend, sessions); the registry is business truth (pricing, bundles, owners, eval history). Registry entries reference Console agent IDs so the two never disagree.

Once an agent is registered, costed, and eval-scored, packaging it into a new bundle is a metadata change, not a project.

12. 30-60-90 roadmap (v3 — Wichita-first)

Phase	Work	Definition of done
Days 0–7 Wichita sprint	§8 runbook end-to-end: Console org live + funded (\$500 credits), triage + quick-audit jobs built, 150 processed, mission-control wall + attendee view wired (demos shipped 6/12), 25 comped deep audits, \$99 code selling	Event executed; ≥30% open their audit; ≥10 deep audits sold; units + tokens per audit logged
Days 8–21 Productize the proof	Webhook bridge production-grade (order form → session → QA queue → Keap + BlitzAdmin). Deep Audit live at \$149. Eval set formalized. Registry v1: frontmatter standard, registry.json generator, dashboard regenerated from it (§11)	A stranger pays \$149 and gets a reviewed PDF in 24h; dashboard reads only from the registry
Days 22–60 Cut COGS 3x, add recurring	Staged pipeline (collectors → Haiku batch → Fable 5 Batch synthesis) → deep audit ≤\$6. Site Builder serves Spotlight orders. Cron refreshes for subscribers. Launch Marketing-Manager + 2 trade bundles. Ahrefs upgrade decision from logged data. Agents self-update their registry entries on every run	Pipeline audit ≤\$6 at eval-equal quality; 3 bundles live; refreshes automated; 239-task backfill complete
Days 61–90 Scale + harden	Multiagent fan-out for 200+ bursts; overlay template so Dylan/Cam ship bundles solo; permission policies across all cert tiers; enterprise volume-discount talk if spend clears ~\$10K/mo	50+/day sustained; new vertical in a day; weekly COGS auto-report

13. Risks, honestly

Managed Agents is beta — behavior can shift between releases. The eval set reruns on every release; skills stay portable (Cowork today, Agent SDK if ever needed).

Wichita is 7 days out from a zero Console org. Mitigation: quick audits need only Sonnet + Batch (boring, reliable); deep-audit comps run the proven Cowork skills in managed sessions; worst case they fall back to Dennis's Cowork at \$20 each and the event still works.

Ahrefs Lite is a real constraint — 100K units/mo, 61K burned. Log units-per-audit this week; upgrade with data, not vibes.

Entry-tier rate limits on a new org — pre-fund day one, wave the batches; caching keeps cached tokens off the meter entirely.

Quality drift at volume — QA quotas non-negotiable; triage routes ambiguous identities to humans rather than auditing the wrong John Smith.

Single-vendor concentration — accepted deliberately; Fable 5's synthesis quality is the moat. Collectors, skills, WordPress are portable.

14. Sources

Managed Agents docs + reference (300 creates/min, 600 reads/min; skills, MCP, vaults, webhooks, cron, multiagent) — platform.claude.com/docs/en/managed-agents · API pricing — [/docs/en/about-claude/pricing](https://docs.en/about-claude/pricing) · Rate limits and tiers (cache-aware ITPM: cached tokens don't count; acceleration limits) — [/docs/en/api/rate-limits](https://docs/en/api/rate-limits) · Team plan (Premium ≈ 6.25x Pro/session incl. Cowork + Code; min 5 seats) — [support.claude.com 9266767 / 12004354](https://support.claude.com/9266767/12004354) · Usage Policy — anthropic.com/legal/aup · **Ahrefs checked live 6/12/26: Lite monthly, 100K units, 61,397 used, resets 6/19** (top-ups: docs.ahrefs.com/api/docs/limits-consumption) · Demo files shipped with this doc: [wichita-mission-control.html](#) + [wichita-attendee-view.html](#) (go-live JSON contracts in file headers) · Internal: DealCon skill library v3; marionarang + julian-david recipes; observed \$20/audit Cowork cost (n=25/day, June 2026). COGS computed from published per-MTok rates; assumptions in §3/§5; re-verify quarterly.