cube

THE AGENTIC FINANCE LAYER

Claude for FP&A

A working guide to finance-grade AI, and how to choose the right model for the job.

CATEGORY

Tactical Guide · Agentic Finance

BY

Cube

FEATURING

Claude by Anthropic

FOREWORD

Two Questions, One Trench Coat

The model question gets all the attention. The context question decides whether any of it works.

Every conversation about AI in finance is two questions wearing one trench coat. The first is the one everyone asks: which model? Fable or Opus, big or fast, this vendor or that one. It fills the keynotes and the LinkedIn feeds. The second is the one that decides whether any of it works: what does the model know? Where do its numbers come from, who governs them, and can you trace an answer back to a transaction when the audit committee asks?

The industry has spent two years obsessing over the first question and ignoring the second. The result is a generation of finance teams with world-class reasoning engines pointed at stale CSVs. The model writes a beautiful variance narrative. The narrative is wrong, because the actuals were three versions old, and nobody can prove which version it was.

This guide covers both questions. It is a working manual for Claude in a finance function: the surfaces, the models, the prompting craft, and then the part that separates a demo from a deployment — the context layer that feeds the model governed, decision-ready numbers. Read it with one workflow from your own close in mind. **By the last chapter you should know exactly how to build it.**

> **"The model you choose matters less than the context you feed it.**
>
> — Cube

Table of Contents

CHAPTER 01

The Claude Stack, Explained for Finance

Claude is several products. Knowing which surface fits which finance job saves you a quarter of trial and error.

"We should use Claude" is a sentence that means five different things, because Claude shows up in five different places. Match the surface to the work and adoption gets easy. Mismatch it and your team concludes AI doesn't work, when what didn't work was the fit.

The Five Surfaces

The chat app, the agentic desktop, the model in your spreadsheet, the coding agent, and the API are the same intelligence with different job descriptions. **Claude.ai** is the front door for ad hoc work, and Projects give a conversation durable context. **Claude Cowork** completes multi-step file assignments end to end. **Claude in Excel** reads and works the open workbook. **Claude Code** writes and maintains automation from plain English. **The API** embeds the model into software, including your vendors' products.

Surface	Built for	Best FP&A fit
Claude.ai	Conversational analysis with durable Project context	Ad hoc analysis, drafting, document review
Claude Cowork	Agentic completion of multi-step file tasks	Reconciliations, variance packages, recurring assembly
Claude in Excel	Reading and working the open workbook	Formula building and auditing, model explanation
Claude Code	Writing and maintaining automation from plain English	Data pipelines, scheduled reporting, transformations

Surface	Built for	Best FP&A fit
The API	Embedding the model into software	Evaluating the AI inside your finance tools

BEST PRACTICE

Pick one surface per use case and standardize. Ad hoc analysis in Claude.ai Projects, spreadsheet work in Claude in Excel, recurring automation in Claude Code. Teams that let every analyst freelance across surfaces end up with five versions of every workflow and no institutional memory.

The Catch

Here is the catch, and it is the thesis of this guide. Every one of these surfaces is a brilliant reasoner staring at whatever you happen to show it. Paste a stale export and it reasons brilliantly about stale numbers. The surfaces solve where you meet the model. They do not solve what the model knows. Hold that thought.

CHAPTER 02

The Model Lineup, and How to Pick

Anthropic ships a family of models. Treat them like a team with different seniority levels, not a single product.

The Claude 5 generation arrived with a new top tier, and with it a new failure mode: reaching for the biggest model for everything. Model selection is a resourcing decision, the same one you make when you staff an analysis. You do not put your most senior person on data entry, and you do not hand the board narrative to an intern.

Four Seniority Levels

FLAGSHIP

Fable 5

The Mythos-class top tier, above Opus. For work where depth compounds: multi-document board narrative, long-horizon scenario reasoning. If the output drives a capital decision, Fable earns its cost.

SENIOR

Opus 4.8

The senior workhorse: deep reasoning for complex recurring work. Judgment-heavy reconciliations, plan-versus-plan analysis, technical accounting research. Most teams' hardest weekly tasks land here.

VOLUME

Sonnet 4.6

The volume player: fast, capable, and the right default for the bulk of FP&A work. Flux commentary at scale, drafts, and data prep.

SPRINTER

Haiku 4.5

The sprinter: tagging, extraction, and routing inside pipelines, where speed and cost matter and a wrong answer is correctable in Slack.

Model	Tier	FP&A jobs	Blast radius
Fable 5	Flagship	Board narrative, long-horizon scenarios	Board-facing

Model	Tier	FP&A jobs	Blast radius
Opus 4.8	Senior	Judgment-heavy reconciliations, technical research	CFO-facing
Sonnet 4.6	Volume default	Flux commentary at scale, drafts, data prep	Internal drafts
Haiku 4.5	Sprinter	Tagging, extraction, routing in pipelines	Correctable in Slack

WATCH OUT

Model names and versions move fast. This lineup is current as of mid-2026; before you hard-code a model into a workflow, check Anthropic's documentation for the latest versions and pricing. Build workflows so the model is a swappable setting, not a load-bearing assumption.

AI INSIGHT

The capability gap between tiers shows up most in ambiguity, not arithmetic. Every model can sum a column. Ask "what should worry me in this forecast?" and the tiers separate fast. Spend up when the question is open-ended; spend down when it is defined.

Route by Blast Radius

A rule of thumb for the budget conversation: match the model to what a wrong answer costs. Spend up as the blast radius grows; spend down when a mistake is cheap to catch.

CORRECTABLE IN SLACK

Haiku / Sonnet

A wrong answer costs a quick correction. Route high-volume, low-stakes work here: tagging, extraction, data prep, first drafts.

CFO-FACING

Opus

A wrong answer reaches the CFO. Judgment-heavy reconciliations and technical accounting research earn the senior workhorse.

BOARD-FACING

Fable + review gate

A wrong answer reaches the board. Run the flagship, and add a mandatory human review gate on top.

And now the closing argument this guide exists to prove: a Haiku connected to governed, current, traceable numbers will beat a Fable reasoning over last month's export.

CHAPTER 03

Context Engineering, or Prompting Like an Analyst

The skill people call prompting is mostly the discipline of supplying context. Learn it manually first; automate it later.

Watch a great analyst brief a new hire and you will see every prompting technique that matters: here is the question, here is the data, here is what the acronyms mean, here is what good looks like, flag your assumptions. Claude responds to exactly the same briefing discipline. Teams getting mediocre output from frontier models are almost never hitting a capability ceiling. They are giving a senior-level reasoner an intern-level briefing.

THE BRIEFING FORMULA

DELIVERABLE + CONTEXT + DEFINITIONS + ASSUMPTIONS SHOWN

Every technique in this chapter is one of these four, done deliberately.

The Four Disciplines

DISCIPLINE 1

Give it a job description

"Analyze this variance" produces a tour. Name the deliverable, the audience, and the decision the output feeds, and you get an analysis.

DISCIPLINE 2

Supply the context

The model doesn't know your fiscal calendar, chart of accounts, or that "the platform team" is cost centers 4100-4140. Every gap is a silent wrong assumption.

DISCIPLINE 3

Define your terms

"Committed spend," "bookings," "active customer." Pin the definitions the way you would for a new hire, so the model counts what you count.

DISCIPLINE 4

Iterate like a review

The first output is a first draft. "Tighten to three sentences," "re-cut by region," "stress the downside." Each turn is a markup.

Explain the $340K unfavorable variance in S&M. Separate one-time from run-rate drivers and tell me what it implies for the Q3 forecast. Cost centers 4100-4140 are the platform team. Flag your assumptions.

CLAUDE

Assumptions first: I'm treating the $120K conference sponsorship as one-time and the two May AE hires as run-rate. On that basis, $180K of the variance recurs, roughly $60K per month against the Q3 plan. The Q3 forecast absorbs it only if the planned July marketing pause holds.

ASSUMPTIONS SHOWN | ONE-TIME VS RUN-RATE | Q3 IMPLICATION

BEST PRACTICE

Build a team prompt library for recurring workflows: the monthly flux prompt, the board-narrative prompt, the contract-review prompt, each with the context block already written. In Claude, store these as Project instructions or skills so they apply automatically, and new analysts inherit the team's accumulated judgment on day one.

WATCH OUT

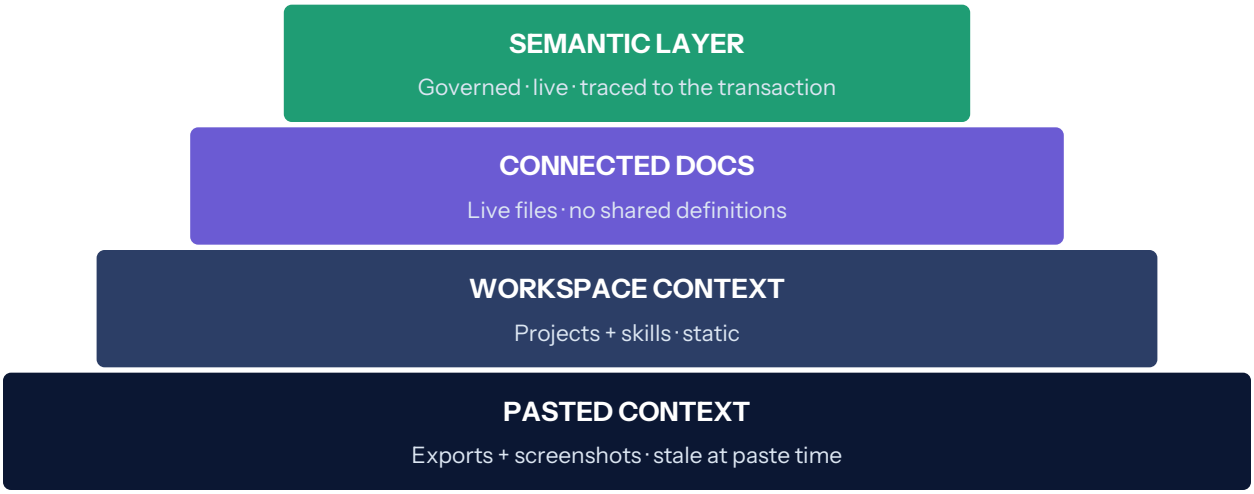
Never paste what you cannot govern. Sensitive comp data, material nonpublic information, customer PII — the question to ask before any paste is "would I email this file to an external party?" If the answer is no, the data needs a governed path, not a chat window.

CHAPTER 04

From Pasted Context to a Context Layer

There is a maturity curve for how AI gets its finance context. Most teams are on step one and don't know steps two through four exist.

Every finance team using AI sits somewhere on a four-stage curve, and the stage you occupy predicts your results better than the model you bought. The curve runs from pasting files into a chat window to a governed semantic layer any model can query live. Each stage up cuts manual work, raises trust, and widens what AI can safely touch.



The Four Stages

Stage 1 — Pasted context. Exports, screenshots, copied tables. It demos brilliantly and decays immediately: the numbers are stale at paste time, two analysts paste two different versions, and nothing is traceable afterward. Dangerous as an operating model, because it manufactures shadow data — ungoverned copies of financial truth scattered across chat histories.

Stage 2 — Static workspace context. Projects and skills. The planning calendar, accounting policies, and the messaging on how you talk about the numbers, uploaded once and applied to every

conversation. Real infrastructure every team should build. Its limit is the word static: documents describe your business; they do not contain your live numbers.

Stage 3 — Connected documents. Claude reads source files where they live — Drive, Slack, internal wikis — so context stays current without re-uploading. A genuine step up for narrative and policy context. But a connected spreadsheet is still a spreadsheet: no shared definitions, no dimensional structure, no guarantee that "revenue" here means "revenue" there.

Stage 4 — The semantic layer. A governed layer above your source systems — GL, HRIS, CRM, billing — holding what a spreadsheet cannot: shared metric definitions, dimensional hierarchies, access controls, and lineage from every figure back to the transaction. When AI queries it, it gets the number, the meaning of the number, and the provenance of the number, current as of now. That bundle is what makes data decision-ready.

Stage	What it adds	Where it stops
1 · Pasted context	Fast, zero setup, demos well	Stale at paste time, untraceable, breeds shadow data
2 · Workspace context	Durable policies and definitions	Static; describes the business, not the live numbers
3 · Connected docs	Current files, no re-uploading	Still spreadsheets; no shared definitions or lineage
4 · Semantic layer	Number, meaning, and provenance, live	The destination: governed and query-ready

> **DATA PRINCIPLE**
> Decision-ready data is the dividing line between AI that demos and AI that deploys. And notice what Stage 4 implies about architecture: the layer has no required front door. Claude queries it, your spreadsheet queries it, your planning UI queries it. The value lives in the layer, and any surface can deliver it.

CHAPTER 05

MCP: The Wire Between Model and Layer

Model Context Protocol is how Claude queries your numbers live. Twenty minutes of setup replaces every paste you would ever do.

Chapter 4 made the argument for a context layer. This chapter is the wiring diagram. Model Context Protocol lets Claude connect to external systems as tools: query them, retrieve from them, act through them, live in the conversation. It is the difference between describing your numbers to the model and handing the model a keycard.

What MCP Is, Minus the Jargon

An MCP server is a doorway a system exposes; Claude is a client that can walk through it. Connect a server in Claude's settings once, and from then on any conversation can call it: you ask a question, Claude decides it needs data, queries the connected system, and reasons over what comes back, current as of the moment you asked. No export, no paste, no version roulette.

STEP 1

You ask

"How did Q2 opex land against plan, and which departments drove the miss?"

STEP 2

Claude reasons

The model decides what data it needs and queries the connected layer live.

STEP 3

The layer governs

Definitions, entitlements, and dimensional structure shape what comes back.

STEP 4

Answer with lineage

Figures arrive current and traced: these accounts, these transactions, as of now.

The drill-down is the killer feature. Summary numbers are where AI trust goes to die, because a summary invites the question "says who?" When the answer comes from a layer with lineage, "says who" has an answer: this figure, from these accounts, from these transactions. The auditability you built into the layer travels with every AI answer drawn from it.

CLAUDE + THE CUBE MCP SERVER, LIVE

How did Q2 opex land against plan, and which departments drove the miss?

CLAUDE

Q2 opex closed at $6.24M against a $5.81M plan, $430K unfavorable. Three departments carry the miss: S and M +$340K, R and D +$65K, G and A +$25K. Every figure below is traceable to the GL.

CUBE · GL 6000-6900 | Q2 ACTUALS VS PLAN | AS OF TODAY

Re-run it excluding the one-time legal accrual.

CLAUDE

Excluding the $85K accrual in G and A, the miss narrows to $345K and G and A lands $60K favorable. S and M remains the driver.

RE-QUERIED LIVE | 1 EXCLUSION APPLIED

AI INSIGHT

MCP inverts the build-vs-buy math on AI in finance. Before MCP, getting AI to know your numbers meant a custom integration project per tool. Now any MCP-capable model can connect to any MCP server. Connect the layer once; every Claude surface — chat, Cowork, Code — inherits it.

CUBE

The Cube MCP Server puts the Agentic Finance Layer one connection away from Claude. Actuals, plans, scenarios, and dimensions become queryable in plain English, every insight maps back to the GL, and the access controls you set in Cube govern what any conversation can see. This is live today; the workflows in the next chapter run on it.

CHAPTER 06

Workflows That Work Today

Five builds with the full recipe — model, surface, and context source. Steal them in this order.

Everything before this chapter was ingredients. Here are five dishes: real FP&A workflows running in finance teams now, each specified the way you would actually deploy it. Notice the pattern — the workflows that survive contact with a real close all have governed context underneath. The ones that quietly die were built on paste.

The Five Builds

- **Month-end flux commentary.** Sonnet · Claude.ai Project or Cowork · layer via MCP. The model pulls actuals against plan by cost center, drafts variance explanations to your materiality thresholds, and flags items needing human narrative. First-draft pass drops from days to an hour.
- **Board narrative.** Fable · Claude.ai Project · layer plus prior decks. Depth compounds and the output is board-facing, so it runs on the flagship with a mandatory review gate.
- **Scenario Q&A.** Fable · Claude.ai · scenarios in the layer. "If we pull the hiring class to Q1 and churn runs 100bps worse, where does runway land?" answered live, in the meeting. Planning stops being a request queue.
- **Headcount reconciliation.** Opus · Cowork · HRIS and plan via the layer. Plan versus actual roster, starts, terms, transfers. Opus judgment matters because the discrepancies are ambiguous by nature; Cowork produces the exception list.
- **Recurring reporting automation.** Sonnet via Claude Code · scheduled · layer via MCP. The weekly KPI snapshot, the Monday metrics email. Claude Code writes and maintains the pipeline; the layer guarantees the numbers. AI stops being a tool you use and becomes infrastructure you own.

WHAT THE ANALYST OPENS · MONTH-END FLUX, OPEX BY COST CENTER ($K) — ILLUSTRATIVE

A	B	C	D	E
Cost Center	**Actual**	**Plan**	**Var**	**Draft Commentary**
S and M	1,840	1,500	+340	One-time event spend $120K; 2 AE hires recur ~$60K/mo
R and D	2,265	2,200	+65	Cloud costs tracking new usage tier; recurs
G and A	1,025	1,000	+25	Legal accrual $85K one-time; offset by open role
CS	1,110	1,110	0	On plan; no commentary needed

DRAFT COMMENTARY BY CLAUDE · FIGURES GOVERNED BY CUBE VIA MCP · EVERY LINE TRACES TO THE GL

Workflow	Model	Surface	Context source	Human gate
Flux commentary	Sonnet	Claude.ai / Cowork	Layer via MCP	Edit pass
Board narrative	Fable	Claude.ai Project	Layer + prior decks	Mandatory review
Scenario Q&A	Fable	Claude.ai	Scenarios in layer	Live, in-meeting
Headcount recon	Opus	Cowork	HRIS + plan via layer	Exception review
Recurring reporting	Sonnet	Claude Code	Layer via MCP	Scheduled + spot-check

CUBE

These five patterns are what Cube's FP&Agents productize — The Data Manager, The Analyst, The Planner, and The Business Partner run finance-grade versions of these workflows on the Agentic Finance Layer, with the governance built in rather than bolted on.

CHAPTER 07

Buying FP&A Software in the AI Era

Most FP&A software evaluations running today use last era's scorecard. The AI era changes what you are actually buying.

Somewhere in your inbox is an FP&A software evaluation built on the old questions: better dashboards, slicker UI, longer feature list. Those questions made sense when the interface was where the value lived. Frontier models just moved the value. The decision that matters is no longer which application your team logs into. It is where your numbers live, how they are governed, and whether the models can reach them.

The Walled Suite Was Built for a World That Ended

The monolithic planning suite made one bet: own the data, own the interface, own the workflow, and charge for the bundle. The walls were the moat. That architecture now works against you twice over. Reasoning comes from models that improve every quarter, delivered in whatever surface the user already occupies — and a vendor's in-house model against Fable is not a fair fight. Meanwhile the walls that once locked users in now lock the models out: your numbers, sealed inside an application the rest of your AI stack cannot query.

THE OLD BET

The Walled Suite

Data sealed inside one application. An in-house assistant racing the frontier and losing. Value concentrated in an interface your team has to log into. Every wall that held users in now holds models out.

THE NEW BET

The Headless Layer

A governed semantic layer any surface can deliver: spreadsheet, chat, any MCP-capable model. Value concentrated in definitions, lineage, and controls. Every model generation is a free upgrade.

The New Scorecard

- **Is there a real semantic layer** — shared definitions, dimensional structure, lineage to the transaction — or just a database behind a UI?

· **Is it headless?** Can the data and intelligence be delivered outside the vendor's own application, into the spreadsheet and tools your team already uses?

· **Can a model you choose query it** — through an open standard like MCP — or can only the vendor's built-in assistant touch the data?

· **Does it connect bi-directionally to the spreadsheet,** so the place finance actually works is a first-class surface rather than an export target?

· **Do entitlements and auditability live in the layer,** so they apply to every surface and every model automatically?

> **WATCH OUT**
> Beware the bolted-on chat box. Every legacy suite now demos "AI features," and a chatbot stapled to a walled architecture is the old model wearing a costume. The tell: ask where the numbers come from and whether a model you choose can query them directly. Architecture answers that question; adjectives don't.

CHAPTER 08

Governance: What Makes AI Finance-Grade

Finance has the strictest correctness standard in the company. Your AI stack has to meet it, and the layer is where it does.

Marketing can tolerate an AI hallucination; the cost is an awkward email. Finance cannot; the cost is a restatement, a busted covenant calculation, or a board decision made on a wrong number. "Finance-grade" is a specific bar: every figure traceable, every access controlled, every output reviewable. None of it requires a perfect model. It requires honest architecture around the model.

PILLAR 1

Traceability

Every AI-surfaced figure answers "says who?" — which system, which accounts, which transactions, as of when.

PILLAR 2

Access control

Entitlements enforced in the data layer, where they cannot be talked around by a clever prompt.

PILLAR 3

Review gates

Human sign-off scaled to blast radius, written down as policy rather than left to habit.

PILLAR 4

No shadow data

The governed path made faster than the paste, so untracked copies stop being created.

Traceability is the foundation. Every number an AI presents should answer "says who?" — which system, which accounts, which transactions, as of when. This is why the context layer is the governance answer and not just the productivity answer: trace to the transaction is a property of the layer, and every model that queries it inherits the audit trail. When Claude queries through a governed layer, it sees exactly what the asking user is entitled to see.

Human gates scale with blast radius. Codify the Chapter 2 rule as policy: define which outputs are draft-only (anything board-facing, anything external, anything feeding a filing) and which can flow without review. Write it down — an unwritten review policy is a review lottery. And kill shadow data on sight: make the governed path the easy path, because prohibition just drives pasting underground. When the MCP connection answers faster than an export-and-paste, the shadow copies stop being made.

BEST PRACTICE

Run the four pillars alongside the Chapter 7 scorecard in every vendor evaluation, as pass/fail questions: Can every AI-surfaced figure trace to source? Are entitlements enforced at the data layer? Where are the human gates? Does the workflow create ungoverned copies? A vendor who answers with adjectives instead of architecture has answered.

DATA PRINCIPLE

Govern the layer once, govern every model forever. Controls bolted onto individual AI tools break the moment a new tool appears. Controls in the layer apply to every surface and every model that queries it, automatically, including the ones you haven't adopted yet.

CHAPTER 09

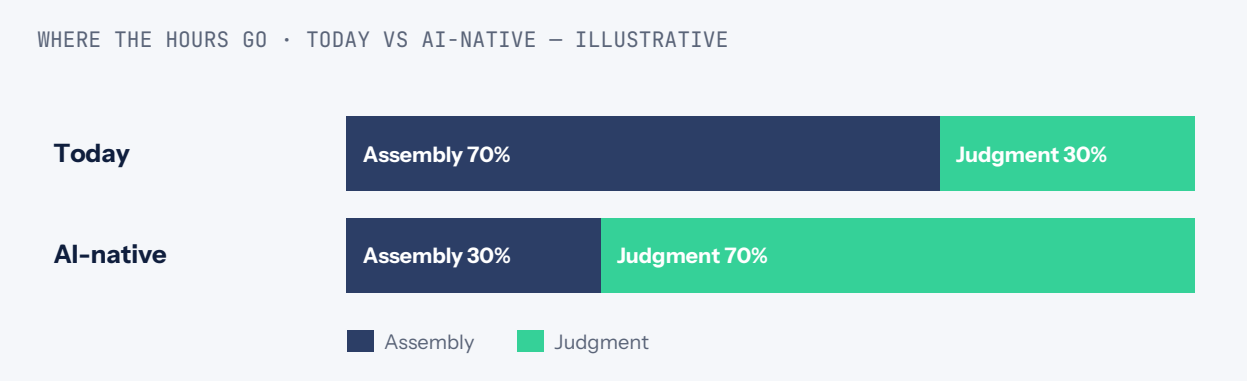
The AI-Native Finance Function

When governed context meets capable models, the shape of finance work changes. Here is where it is going.

Strip an FP&A calendar down to its parts and you find a confession: most of the function's hours go to assembling context — pulling actuals, reconciling versions, formatting the deck — and a thin remainder goes to the judgment the function exists to provide. That ratio was never a choice. It was a constraint, imposed by how long assembly took. The constraint just dissolved.

The Work Inverts

In an AI-native finance function, the layer assembles, the model drafts, and the human decides. The analyst's week shifts from building the variance report to interrogating it; from formatting the scenario to choosing between scenarios. This is the climb up the Strategic Finance Hierarchy of Needs that finance leaders have promised their teams for a decade — and the first time the hours to make it have actually existed.



WHERE HOURS GO TODAY

The Assembly-Heavy Team

Pulling actuals. Reconciling versions. Formatting decks. Answering data requests. Judgment squeezed into whatever time the assembly leaves over.

WHERE HOURS GO NEXT

The Judgment-Heavy Team

Interrogating drafts. Choosing between scenarios. Partnering in the room where decisions happen. Assembly handled by the layer, drafting by the model.

The team reshapes: fewer report-builders, more business partners. The hiring profile tilts toward people who ask sharp questions of good data — operators, skeptics, storytellers. The training priority becomes context engineering and review judgment: knowing what to ask, and knowing when an answer smells wrong. Headcount leverage stops meaning "do more reporting with fewer people" and starts meaning "put a finance brain in every decision."

What does not change: judgment, accountability, and the signature. The CFO still signs the numbers, and no architecture diagram moves that responsibility. AI-native does not mean the human steps aside. It means the human signs with better information, sooner, with the lineage one click away.

> **"The layer assembles. The model drafts. The human decides.**
>
> — Cube

About Cube

The Agentic Finance Layer — a governed data and AI layer that delivers decision-ready intelligence into every tool finance uses.

Cube is the Agentic Finance Layer: a governed data and AI layer that sits above your source systems and delivers decision-ready financial intelligence into every tool finance uses.

Headless by design, Cube works in your spreadsheet, your chat, and any MCP-capable model, with its own interface available but never required. Built by a three-time CFO, Cube unifies actuals, plans, and scenarios with shared definitions, access controls, and lineage that traces every insight to the source — so the numbers your team, your models, and your board rely on are current, consistent, and provable.

The Data Manager

DATA LAYER · GOVERNANCE

Keeps data connected, mapped, and decision-ready across the GL, HRIS, CRM, and billing, with lineage to the transaction.

The Analyst

VARIANCE · FLUX · AD HOC

Variance, flux, and ad hoc analysis with every figure traced to the source, drafted before the close meeting starts.

The Planner

SCENARIOS · FORECASTING

Scenarios, forecasts, and rolling plans that update as actuals land, queryable live, in plain English.

The Business Partner

DELIVERY · NARRATIVE

Delivers governed answers where decisions happen, from Slack to the spreadsheet to the boardroom.

Decision-ready data, everywhere you work.

Cube is the Agentic Finance Layer built for teams who are serious about FP&A.

cube

THE AGENTIC FINANCE LAYER

cubesoftware.com