AI & BUSINESS

AI Agents in Commercial Teams

What actually works when you put an agent next to an analyst, and where the idea quietly falls apart.

Berry Ostrovsky · 2026-08-15 · 9 min read

EXECUTIVE SUMMARY

- An agent is useful when a task is repetitive, well bounded, and verifiable. Most commercial work fails at least one of those tests before it is redesigned.
- The strongest early use cases sit around the analyst, not instead of them: gathering context, preparing inputs, drafting first passes and checking work.
- Tool access matters more than model choice. An agent that can query the same tables and documents the team uses beats a smarter model working from pasted text.
- Every agent output needs a verification path. If a human cannot check a result in less time than producing it, the agent has added risk rather than capacity.
- Pricing, forecasting and customer-facing commitments should stay decision-support, not automation, until the audit trail is genuinely reliable.

The conversation about AI agents in commercial organizations has moved faster than the evidence. Vendors describe autonomous analysts that monitor performance, investigate deviations and act. Commercial teams, meanwhile, are still trying to get a clean weekly sales file that everyone agrees on.

Both pictures are real. The gap between them is where most agent projects are won or lost. This article is about what an agent can actually carry inside a sales, pricing or category team today, what it should not be asked to carry, and how to structure the work so the output is worth trusting.

What an agent is, in practical terms

Strip away the marketing and an agent is a language model placed in a loop: it receives a goal, chooses among tools it has been given, observes the result, and repeats until it decides it is finished. Anthropic's engineering guidance draws a useful line between workflows, where the steps are fixed in code, and agents, where the model itself directs the path.

That distinction matters commercially. A workflow is predictable and cheap to audit. An agent is flexible and harder to audit. Most commercial tasks that look like agent work are actually workflows with one uncertain step in the middle, and they are better built that way.

The practical question is therefore not "should we use agents" but "which single step here genuinely needs judgment, and what can stay deterministic around it".

The three tests a task has to pass

Before assigning anything to an agent, it helps to run the task through three checks.

- Repetition. Does this happen often enough that automating it repays the effort of specifying it properly? A monthly task done twelve times a year rarely does.
- Boundaries. Can the task be described with a clear start, a clear finish and a defined set of inputs? "Investigate why the region missed target" fails this test. "Assemble the standard deviation pack for the region" passes it.
- Verifiability. Can someone confirm the result quickly and objectively? A summary of a call transcript is checkable in a minute. A recommended list price is not.

Tasks that pass all three are where agents produce real capacity. Tasks that fail verifiability are the ones that quietly damage trust, because errors surface weeks later inside decisions that have already been made.

Where agents earn their place today

In commercial teams, the reliable wins cluster around preparation and review rather than conclusions.

- Context assembly. Pulling together what is known about a customer or category before a meeting: recent performance, open issues, previous commitments, contract terms, competitive moves.
- Data preparation. Reconciling naming conventions, mapping customer hierarchies, flagging suspicious rows, and explaining what changed since the last extract.
- First-pass drafting. Turning an agreed analysis into a structured narrative, a customer email or a slide skeleton, which the analyst then edits.
- Checking. Reviewing a finished deck or model for internal inconsistencies, stale numbers, or claims not supported by the underlying data.
- Research with citation. Gathering public information on a market or competitor with links, so a human can verify each claim rather than trust a summary.

None of these replaces an analyst. Each removes an hour of low-judgment work per instance, and the compounding of those hours is the actual business case. It is unglamorous, and it is where the return currently is.

Where they fail, predictably

The failures are not random. They follow the structure of commercial data.

First, commercial datasets carry meaning that is not in the data. A negative volume line may be a return, a correction, or a data error, and the difference is institutional knowledge. An agent will produce a confident explanation for whichever interpretation its context happens to suggest.

Second, small errors compound across steps. An agent that misreads one hierarchy mapping early in a chain will build every later step on it, and the output will look coherent. Coherence is not correctness, and it is precisely what makes these errors expensive to catch.

Third, anything with a commitment attached, a price, a forecast used for supply, a promise to a customer, has an asymmetric cost profile. The upside of automation is a saved hour. The downside is a margin decision that is difficult to reverse.

Tools beat cleverness

The most common reason an agent underperforms in a commercial setting is not model capability. It is that the agent cannot reach the things the team actually uses: the warehouse, the price list, the trade terms document, the CRM.

This is the problem open standards such as the Model Context Protocol are designed to address, by giving models a consistent way to connect to data sources and tools instead of a bespoke integration per system.

The design implication is straightforward. Invest in a small number of well-described, read-only tools with clear scopes before investing in a more capable model. An average model with an accurate query tool will outperform a stronger model working from pasted spreadsheet fragments, every time.

Designing the verification path

The rule I apply is simple: if verifying the output takes longer than producing it manually, the agent has not added capacity, it has moved effort and added risk.

That makes verification a design input rather than an afterthought. In practice it means agents should return the intermediate evidence, not only the conclusion: the query it ran, the rows it used, the documents it cited, the assumptions it applied.

It also means keeping a written record of what an agent is allowed to touch and how its output is reviewed. Governance frameworks such as the NIST AI Risk Management Framework describe this in terms of mapping context, measuring performance and managing risk over time, and the same logic applies to a three-person commercial team, at a smaller scale.

A realistic operating model

A structure that holds up in practice separates three layers, each with a different level of autonomy.

- Deterministic layer. Extracts, joins, business rules and calculations stay in code or SQL. No model decides what the number is.
- Judgment layer. The agent classifies, prioritizes, drafts and explains, always with its evidence attached.

- Decision layer. A person approves anything that touches price, customer commitments or externally reported figures.

Teams that blur these layers get the failure everyone fears: a plausible narrative built on an unverified number. Teams that keep them apart tend to find agents boringly useful, which is the correct outcome.

How to start without a program

Agent adoption does not need a transformation initiative. It needs one task, chosen well.

Pick something the team does weekly, that everyone dislikes, and where the correct answer is obvious once you see it. Preparing the customer review pack is a good candidate. Setting next quarter's price ladder is not.

Run it in parallel with the manual process for a few cycles, record where it disagrees with the analyst, and fix the tooling rather than the prompt when it does. If after a month the team would notice its absence, expand. If not, the task was the wrong one, and that is cheap information.

The useful framing is not autonomy. It is division of labour. Commercial work is a mix of mechanical steps, judgment steps and accountable decisions, and agents are currently strong on the first two when the boundaries are drawn explicitly.

The teams getting value are not the ones with the most ambitious deployments. They are the ones that were honest about which parts of the job are genuinely verifiable, automated exactly those, and kept a person on everything that carries a commitment.

KEY TAKEAWAYS

- Test every candidate task for repetition, clear boundaries and fast verification before automating it.
- Put agents around the analyst: context, preparation, drafting and checking, rather than conclusions.
- Give the agent proper tool access before reaching for a stronger model.
- Require intermediate evidence, not just answers, so results can be checked in seconds.
- Keep price, forecast and customer commitments under human approval.

SOURCES

- Anthropic, Building Effective Agents (engineering guidance on workflows vs. agents)
- Model Context Protocol, official documentation
- NIST, AI Risk Management Framework (AI RMF 1.0)
- Regulation (EU) 2024/1689, the EU Artificial Intelligence Act

This article describes working practice and general guidance. It is not legal, financial or compliance advice, and it

does not describe the systems or data of any specific employer.