

THE AI WAY

Beyond the Chatbot

Three AI agents I shipped to production that move revenue, and what broke.

Rwitapa Mitra

Director BI @ FabHotels

Founder, The AI Way

AI Dev Day India 2026 · Code, Commerce & Qubits · Noida





Rwitapa Mitra

Director of Business Intelligence

- ▶ **10+ years** leading analytics, now Director of BI @ FabHotels
- ▶ **Founder, The AI Way** · Built for Analysts
- ▶ ex-**Flipkart · PharmEasy · Pilgrim · Mu Sigma**
- ▶ *The person leadership pings at 8am: "why is occupancy down?"*



THE THESIS

Most companies are losing money on AI.

They're treating it like an expensive parlor trick. A chatbot in the corner of the website that wows the board and never touches the P&L.

THE SHIFT

The static dashboard is dead. The win goes to whoever plugs AI straight into a real business bottleneck and moves an actual number.



THE PROMISE

Three agents. One live business. Real P&L.

01

Call-QA Scorer

Score 100% of sales calls against an LLM rubric.

+3% conversion

02

Pricing Agent

Scrape the market, suggest data-backed prices.

250+ updates / day

03

NL-to-SQL Bot

Ask the warehouse in plain English, in Slack.

Self-serve, safely

Each one moved a number. Each one broke in a way the demo never warned me about.



We were auditing 2% of sales calls. And pretending it was 100%.

- Inside sales: a black box of people on phones, selling.
- Quality "checked" by a manager spot-listening to a handful of calls a week.
- Hundreds of agents. So ~2% of calls audited, 98% guessed.
- Conversion leaks stayed invisible until the monthly number. By then the money was gone.



Score every call, automatically.



That's the whole architecture. A lean team, not a research lab.



THE BREAK

The model flagged my best agents as rule-breakers.

A rep said **"I can't promise you that rate today."** The model dinged it for "negative language." That rep was honest, correct, and compliant. The rubric said avoid negative language, and the LLM took it literally. It always will.

+3%

ORDER CONVERSION

THE LESSON

Your prompt is a policy document, not a wish. You're writing law for a literal employee with no judgment. Fix the rubric, not the model.



THE PROMISE

Three agents. One live business. Real P&L.

01

Call-QA Scorer

Score 100% of sales calls against an LLM rubric.

+3% conversion

02

Pricing Agent

Scrape the market, suggest data-backed prices.

250+ updates / day

03

NL-to-SQL Bot

Ask the warehouse in plain English, in Slack.

Self-serve, safely

Each one moved a number. Each one broke in a way the demo never warned me about.



250 prices a day. Set by gut.

- A revenue manager sets ~250 room rates daily. Nine years of good gut, genuinely.
- But gut can't watch the whole market at once.
- A competitor drops their rate at 11pm. Gut finds out tomorrow.
- Gut can't explain itself in a review. And gut takes a sick day.

I didn't want to replace the manager. I wanted to give them eyes on the entire market, in real time.



THE BUILD

Eyes on the whole market.



The agent proposes. The human disposes.

THE BREAK

Scraper drift: pricing rooms on invisible data.

An OTA quietly redesigns its site (*the layout an engineer calls the DOM*). Our scraper looks for the price in the old spot. It's gone. Returns **null**. The agent doesn't crash. It happily prices on nothing, and the number still looks plausible.

250+

UPDATES / DAY

THE LESSON

Bad data that looks like good data is worse than a crash. Every agent that reads the outside world needs a freshness gate: real, recent, sane, or stop.



THE PROMISE

Three agents. One live business. Real P&L.

01

Call-QA Scorer

Score 100% of sales calls against an LLM rubric.

+3% conversion

02

Pricing Agent

Scrape the market, suggest data-backed prices.

250+ updates / day

03

NL-to-SQL Bot

Ask the warehouse in plain English, in Slack.

Self-serve, safely

Each one moved a number. Each one broke in a way the demo never warned me about.



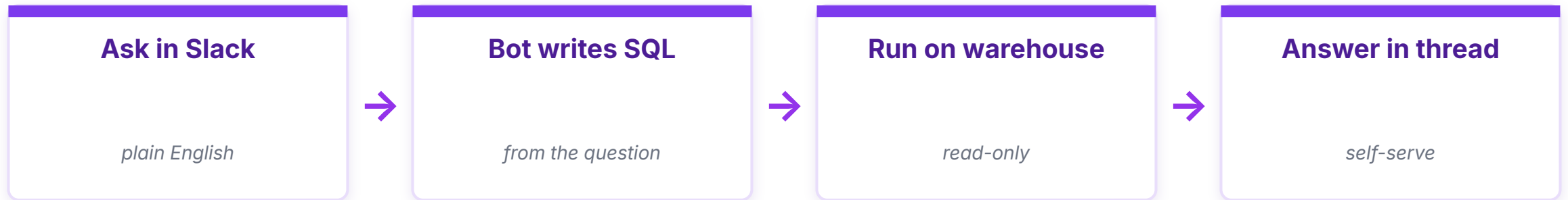
My BI team was the bottleneck on the company's own data.

- Every "quick number" funnels through BI. "Mumbai occupancy last Tuesday?"
- Thirty seconds of SQL. A four-hour queue, because we're human and outnumbered.
- The team becomes a gate on the company's own data.
- So: what if anyone could just ask the database in plain English?



THE BUILD

Ask the database in plain English.



Everyone gets to be their own analyst.



An LLM with an open DB connection is an intern with root access to prod.

Two failure modes. One: ambiguity. "How are bookings doing?" The bot guesses, writes confident SQL, returns a subtly wrong number, and someone takes it into a meeting. Two: one hallucinated **DELETE**, and you're writing a very different update to leadership.

THE TWO GATES

Read-only by default: it physically can't write (architecture, not instruction). Clarify gate: ask back, then show the SQL before running. The human goes from creator to editor.



THE PATTERN

Three failures. The fixes rhyme.

1

Prompt as policy

Write law, with examples, for a literal employee.

from the call scorer

2

Freshness gate

Never act on stale or empty input.

from the pricing scraper

3

Clarify + read-only

Guess nothing. Destroy nothing.

from the SQL bot

Build one AI agent this year? These three gates are the difference between a tool and a liability.



THE WHOLE JOB



**Not one fix was a better model.
Every fix was a guardrail.**

The model is the easy 80%.

Knowing when not to trust it is the whole job.

That's product taste, not ML.



WHY THIS MATTERS

You don't need a Silicon Valley budget.

THE REAL CHANGE

Stop treating AI as a chatbot. Start treating it as an automation pipeline you point at your most expensive bottleneck. I shipped all three as a BI director, lean team, cheap models.



TAKE THE BLUEPRINTS

The prompts behind all three agents. Free.

- The rubric-as-policy pattern
- The freshness gate
- The clarify gate + read-only setup
- The SQL-debug prompts

Go build the paranoid version first.



Scan for the free prompts

theaiway.in/grab/free-prompts

