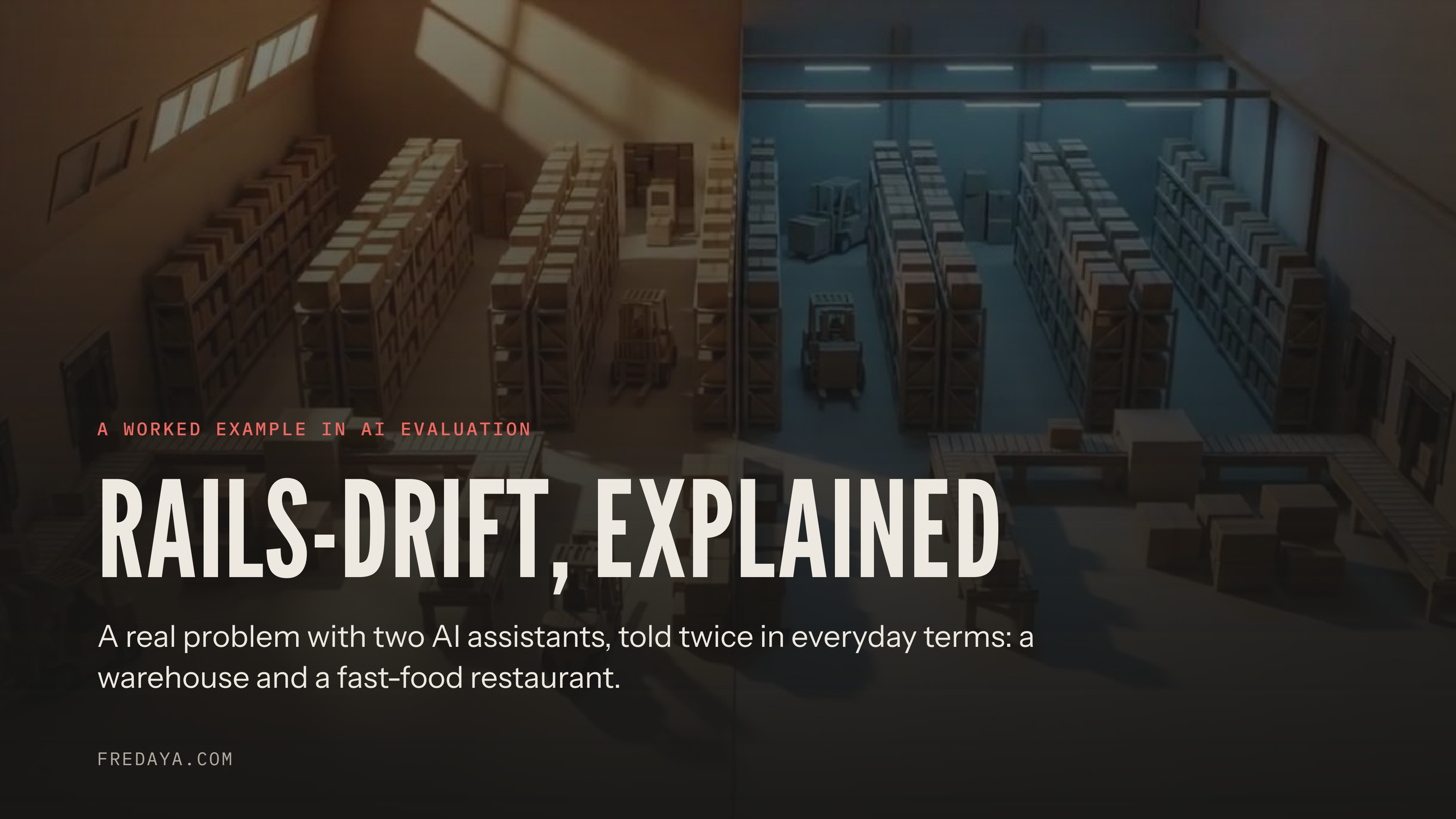




A WORKED EXAMPLE IN AI EVALUATION

RAILS-DRIFT, EXPLAINED

A real problem with two AI assistants, told twice in everyday terms: a warehouse and a fast-food restaurant.

THE REAL USE CASE

TWO AI ASSISTANTS IN SLACK, ONE SET OF WRITTEN RULES.

A support assistant answers a training cohort; an advisor answers clients. For each question a model returns a short result: reply or hand over, how sure it is, a one-line restatement, a reason. Then ordinary code picks one of five outcomes.

Answer

sure enough

Check first

fairly sure, or a long
voice note

Park a draft

not sure enough

Page the operator

a person is needed

Report an outage

the AI service failed

WHAT WENT WRONG

THREE DIFFERENT SITUATIONS REACHED THE OPERATOR LOOKING THE SAME.

A hard question

the model could not answer: this one should reach the operator

A good answer with no confidence number

read as zero, so it was parked for approval

An AI-service outage

paged as if a person needed help





WHY IT MATTERS

FALSE ALARMS COST ATTENTION; SOME MISTAKES REACH PEOPLE.

The operator's attention is the scarce resource. Outages that look like emergencies teach people to ignore alerts.

An unverified answer can reach a real person as if it had been checked.

A real escalation can be lost among the false ones.

WHY MAKE IT A TEST

FIXING WHAT WAS REPORTED IS THE EASY PART.

Coding agents are increasingly asked to maintain systems like this. rails-drift rebuilds the situation from scratch, with no client or production code, and asks an agent to fix it.

11

places where the code breaks the
written rules

3

named in the operator's report

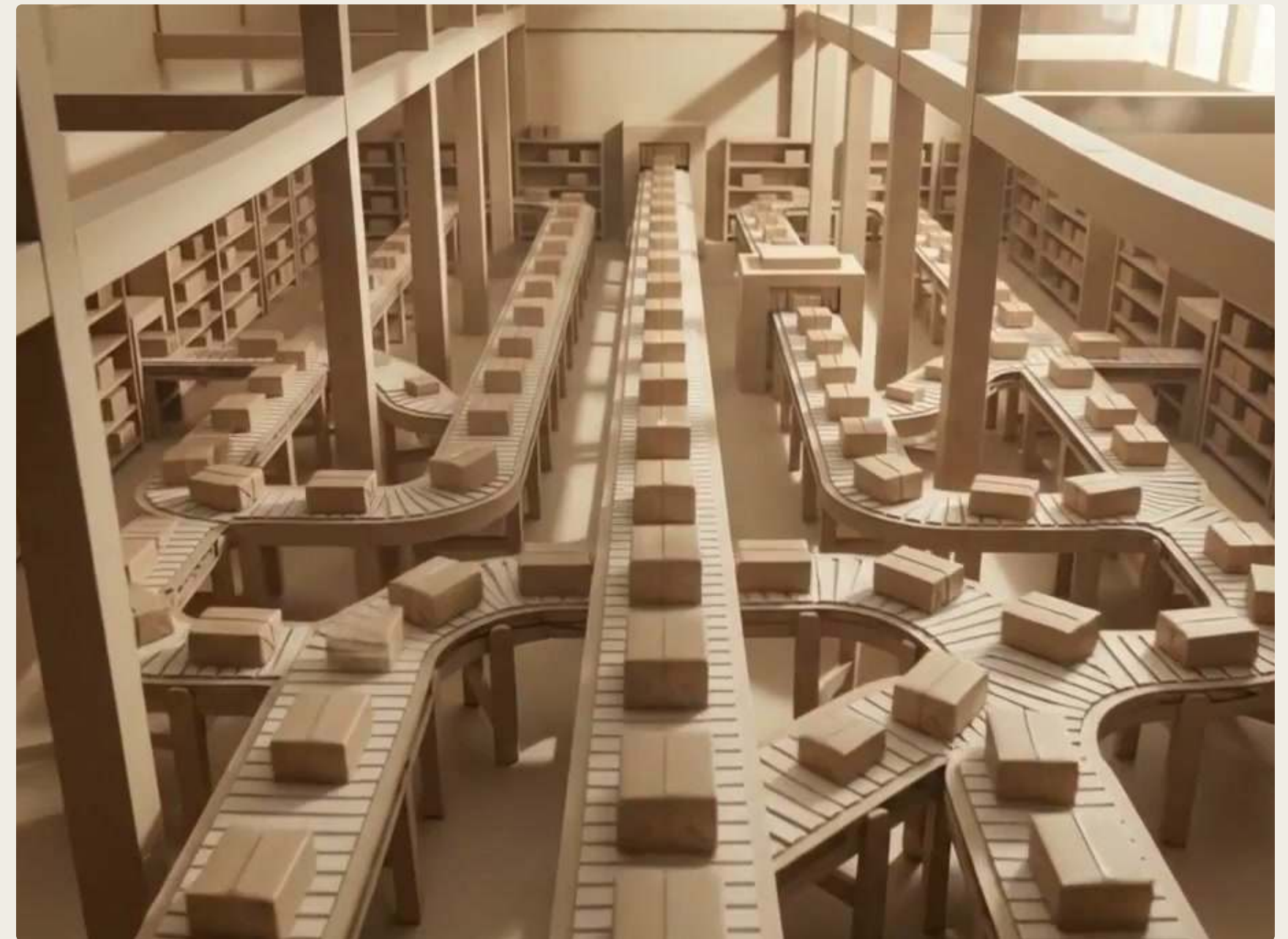
8

found only by reading the rules
against the code

ANALOGY ONE: A WAREHOUSE

TWO SHIFTS, ONE RULEBOOK, COPIES THAT DRIFTED.

Two shifts	the two assistants
Parcels	questions
Operations manager	the operator
Head-office rulebook	the written policy
Limits on the board	the configuration files





THE GRADER, AS AN AUDITOR

WATCH WHAT EACH SHIFT DOES, NOT WHAT IT LOGS.



45

Hidden test parcels

20

Random parcels at the limits

25

Changing the board

10

The lead's own checklist

THE SAFETY CAP

ONE CRITICAL FAILURE CAPS THE SCORE AT 25.

Like a forklift safety violation failing a warehouse audit, however tidy the shelves.

The three critical failures are what the policy exists to prevent: an outage paged as an emergency, a real emergency missed, and an unverified answer sent to a person.



ANALOGY TWO: A FAST-FOOD RESTAURANT

TWO COUNTERS, ONE MANUAL, COPIES THAT DRIFTED.

Front counter and
drive-through

the two assistants

Orders

questions

Store manager

the operator

Operations manual

the written policy

Numbers on the
kitchen board

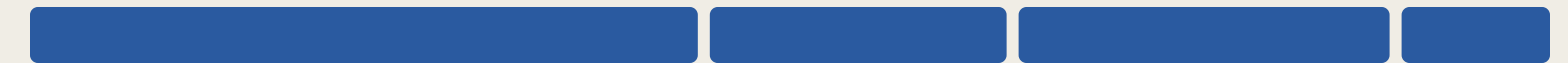
the configuration files





THE GRADER, AS A MYSTERY SHOPPER

TEST ORDERS AT BOTH COUNTERS, THEN CHANGE THE BOARD.



45

Hidden test orders

20

Random orders at the limits

25

Changing the kitchen board

10

The lead's own checklist

THE SAME CAP, IN A KITCHEN

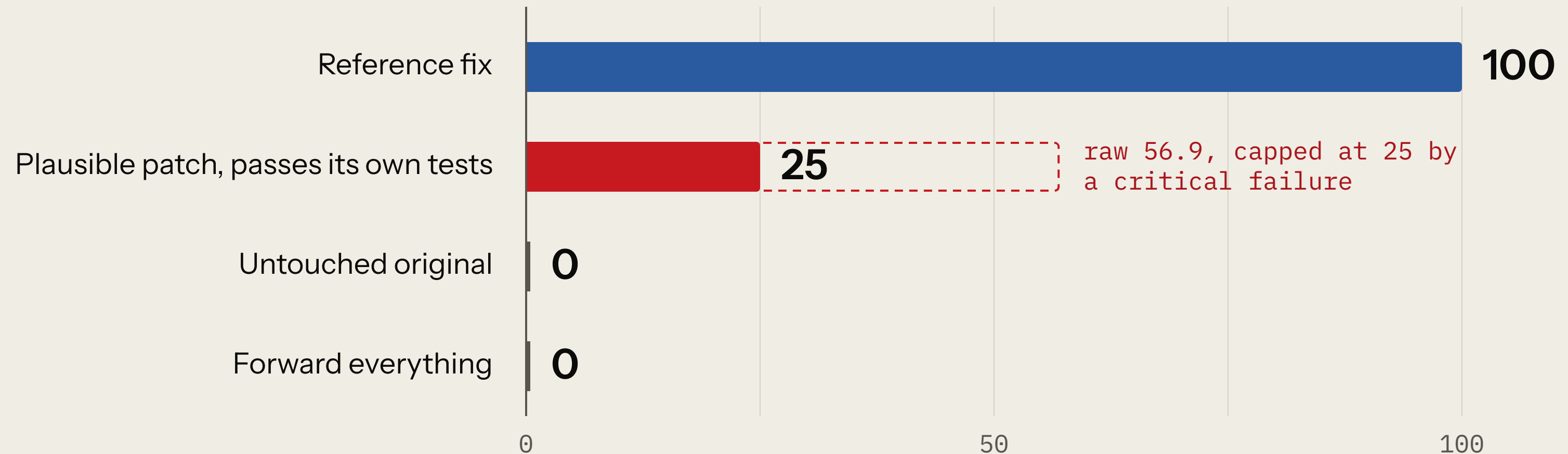
SERVING AN ALLERGY ORDER UNCHECKED FAILS THE INSPECTION.

However clean the floors are, one serious food-safety failure caps the grade at 25 out of 100.



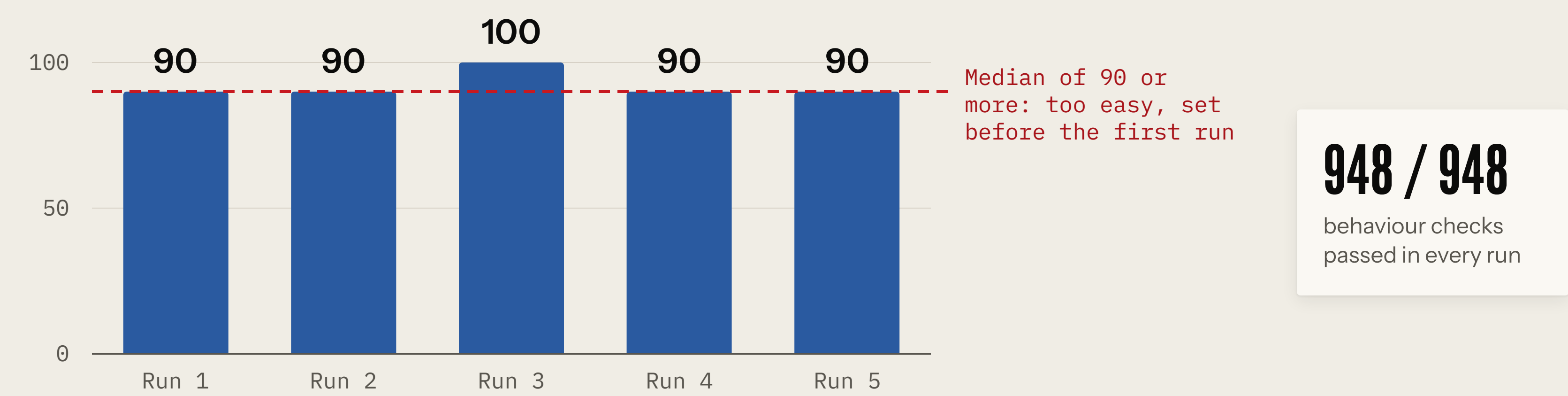
TESTING THE GRADER FIRST

A PATCH THAT PASSES ITS OWN TESTS STILL SCORES 25.



THE FIRST CALIBRATION

FIVE RUNS OF ONE FRONTIER CODING AGENT.



Every run fixed all eleven deviations. The task is too easy at this level, so a longer version follows.

WHERE THE POINTS WENT

EVERY LOST POINT TRACED TO THE INSTRUCTIONS.

460 points earned across five runs

40 lost: the task's wording



of 500 possible

0 lost: the agent's logic

Four runs wrote tests that use a helper their own fix created, so the tests could not load on the original code. In the warehouse: checklists written for a new labelling system, tried in the old warehouse.

The instructions never required it, so the fault was the task's wording, not the agent's. Recorded scores stay as they are.

WHAT IT SHOWS

A GRADER THAT HOLDS UP, ON A TASK THAT IS TOO EASY.

The grader holds

It agreed with the written rules on every check, and its one losing check has a documented cause.

The task saturates

A frontier coding agent fixed every planted deviation in every run.

The limits

Five runs of one agent on one machine; no comparison yet. The grader is public: a worked example, not a hidden benchmark.

THE POINT

**READ THE RULES AGAINST THE
CODE, NOT JUST THE COMPLAINT.**