Gradient Labs

Shipping secure, reliable and high performance AI agents

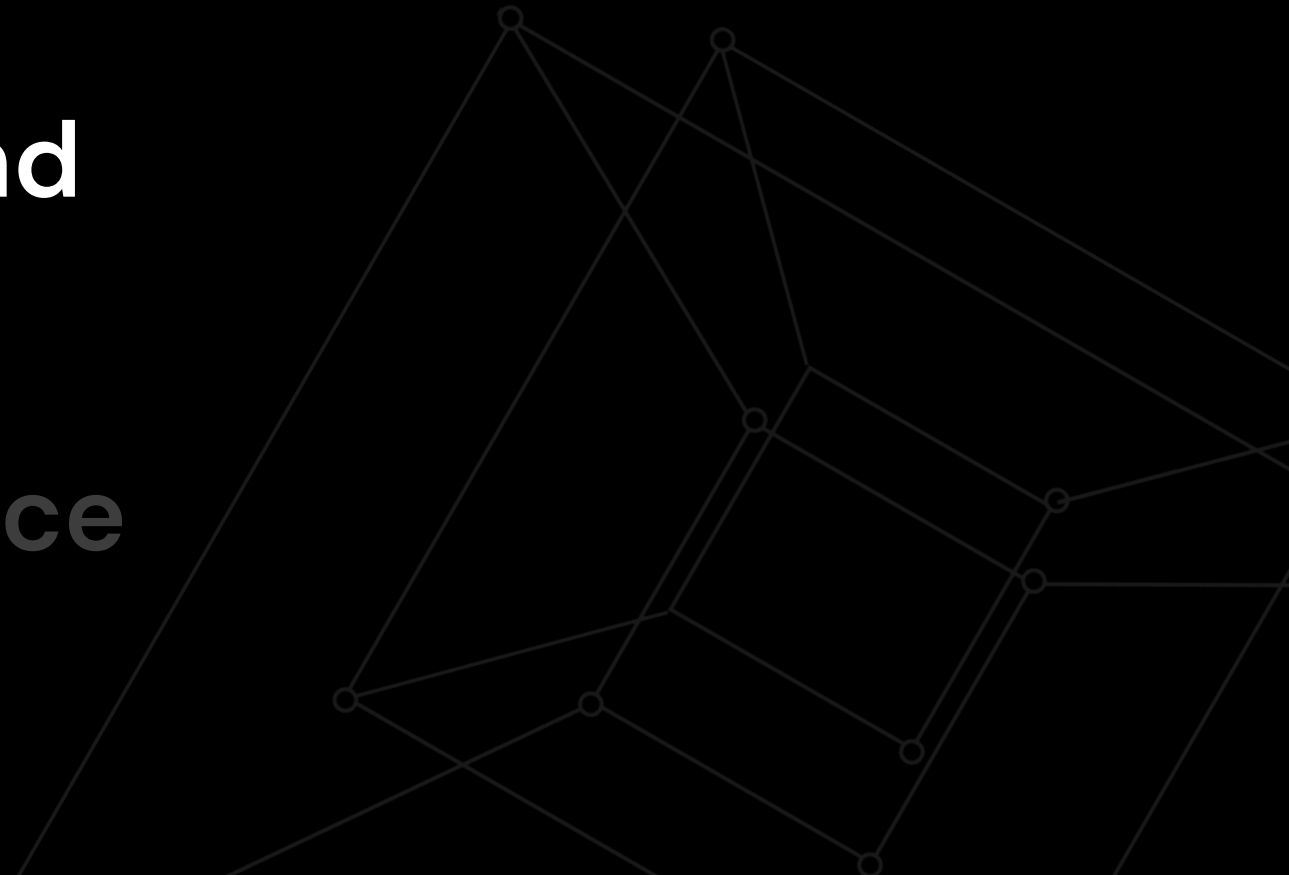
Danai & Devan

Background

Security

Reliability

Performance



Agents as execution harnesses

An agent is not one big prompt.

It is a runtime harness that decides what objective is safe:

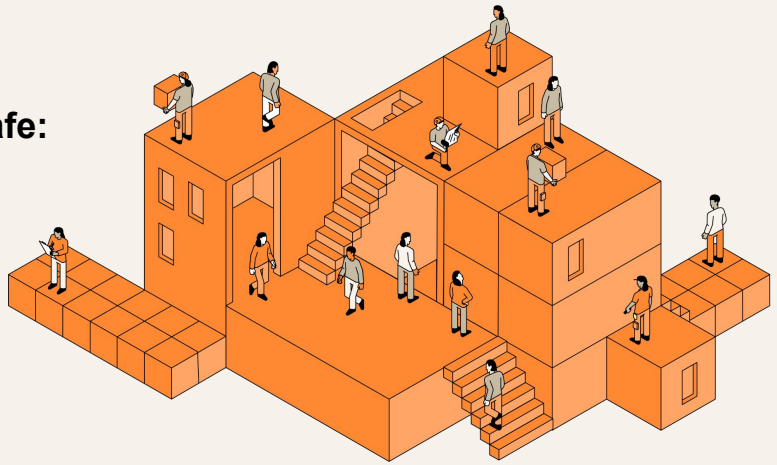
01 what the user is trying to do

02 what context is allowed

03 which tools are available

04 what checks are required

05 when to reply, wait, refuse, or hand off



Control plane + runtime harness

Control plane: what exists and who can use it

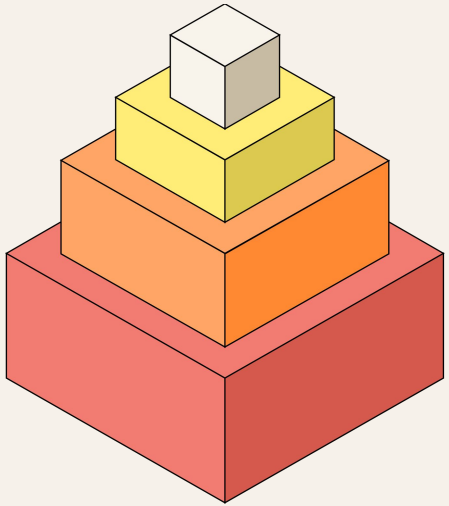
- companies, procedures, resources
- tools, tokens, integrations
- analytics and shared model infrastructure

Runtime: what happens on this turn

- chains, skills, workflows
- preconditions, guardrails, tool orchestration

Procedures set the boundary

Procedures turn business policy into runtime policy.

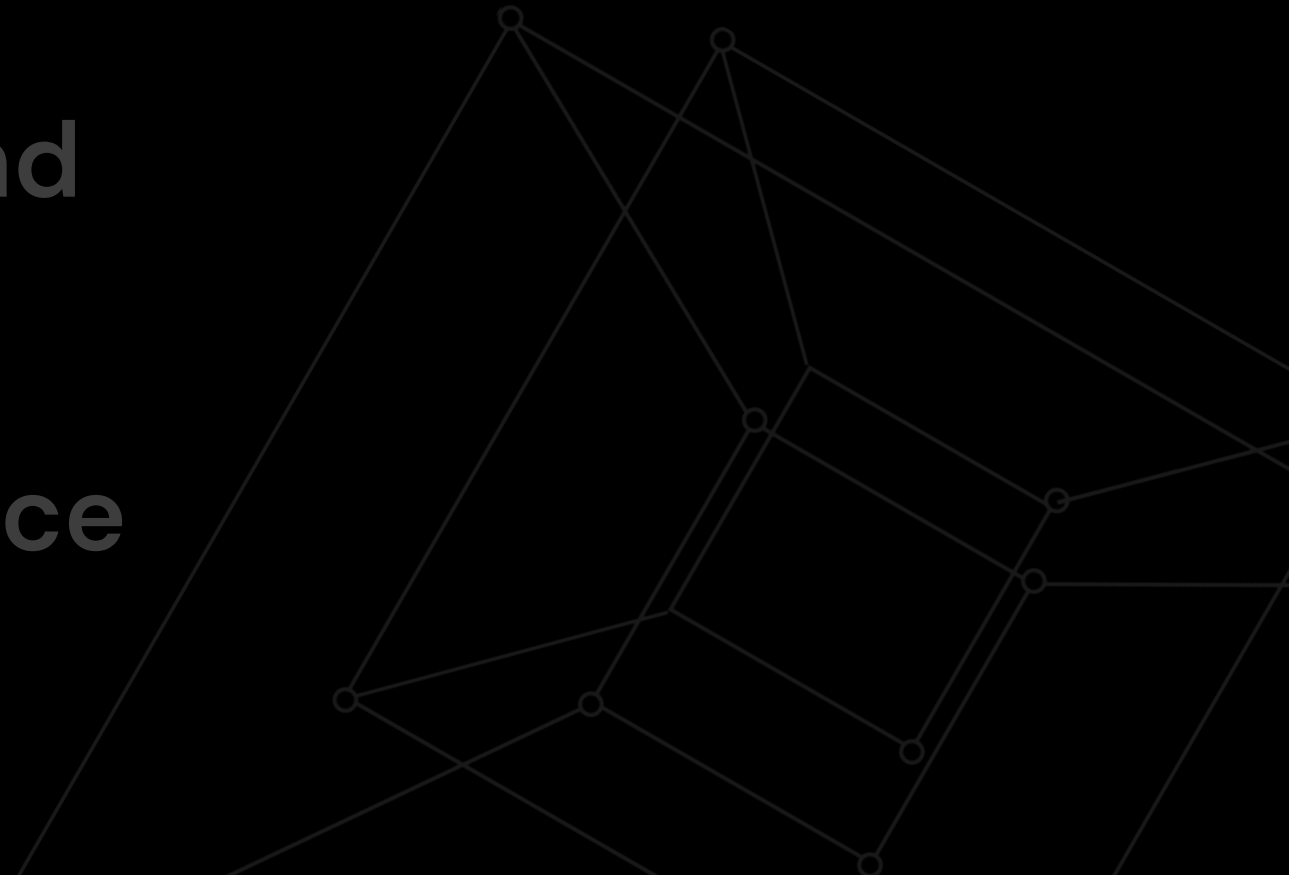


They scope:

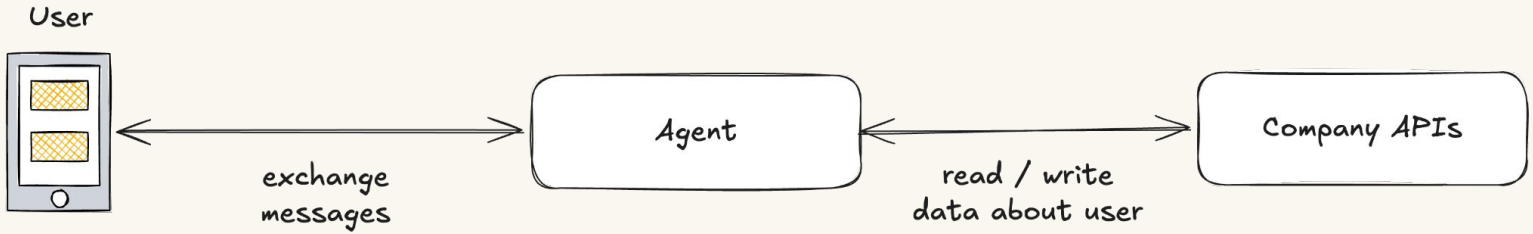
- resources
- tools
- identity requirements
- escalation paths
- relevant guardrails

A complaint, payment plan, callback request and IDV flow should not expose the same context or actions.

Background
Security
Reliability
Performance



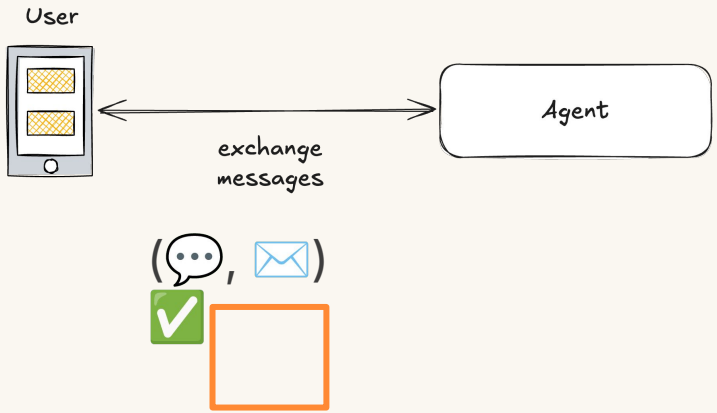
A high-level picture



1. Make sure we're talking to the right person

2. Limit access to reading and writing the minimal data required

Making sure we're talking to the right person



Security questions: what's your…

- Email
- Postcode
- Current balance
- Policy / membership number
- …

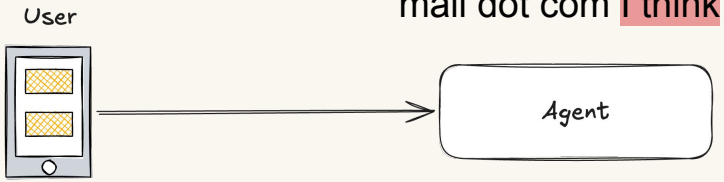
Making sure we're talking to the right person

Ground truth

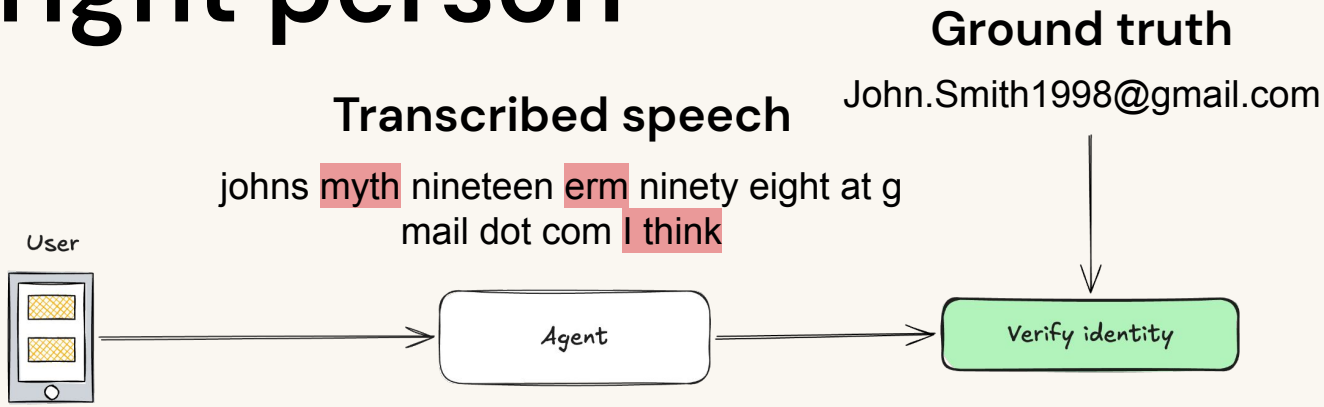
John.Smith1998@gmail.com

Transcribed speech

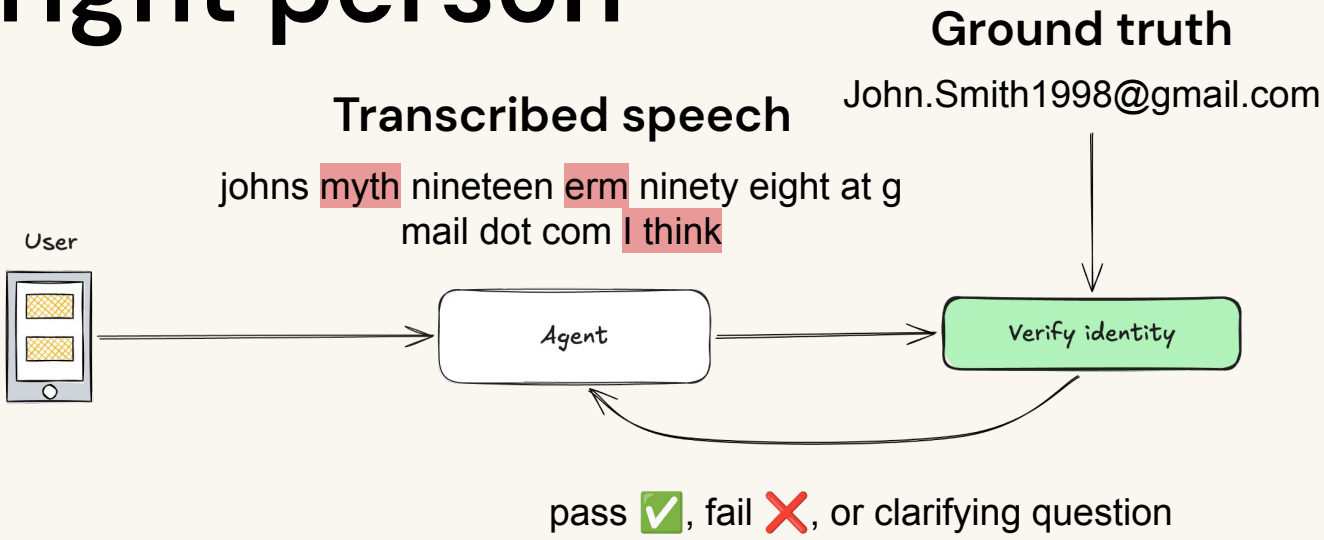
johns myth nineteen erm ninety eight at g mail dot com I think



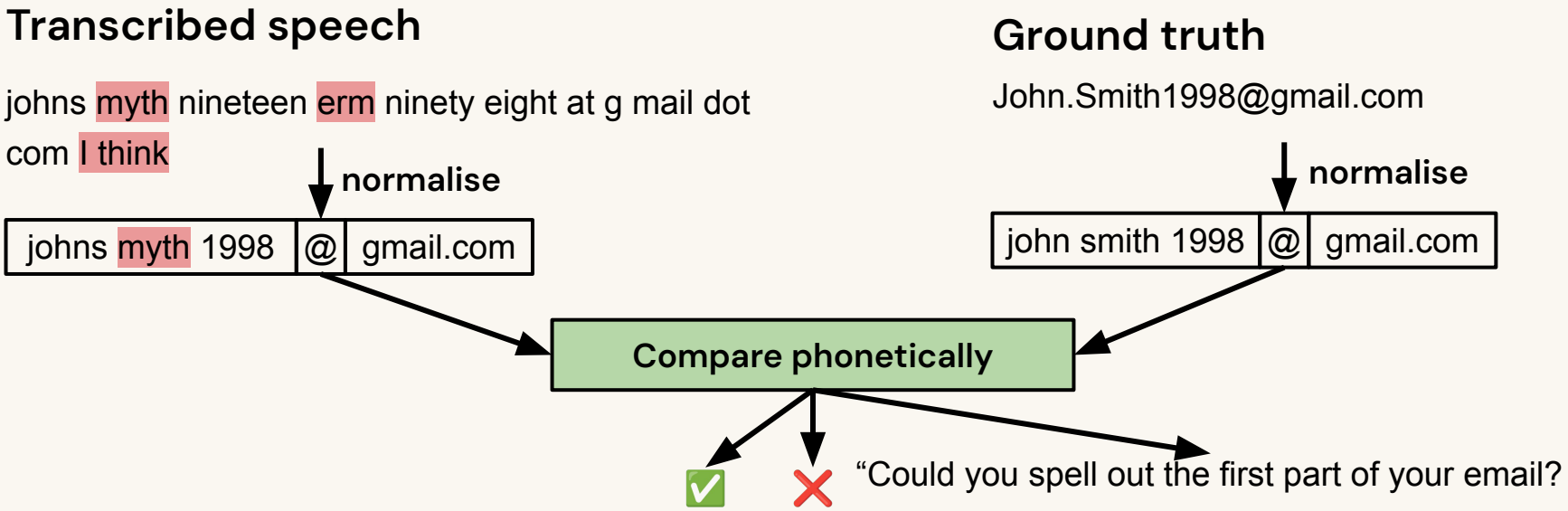
Making sure we're talking to the right person



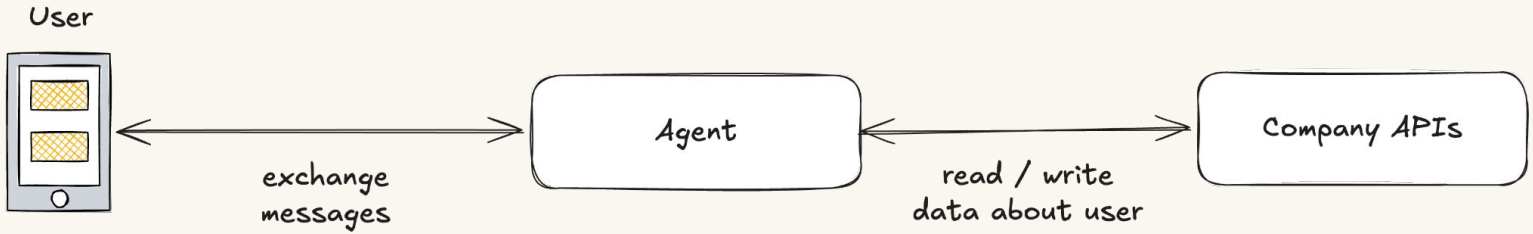
Making sure we're talking to the right person



Making sure we're talking to the right person



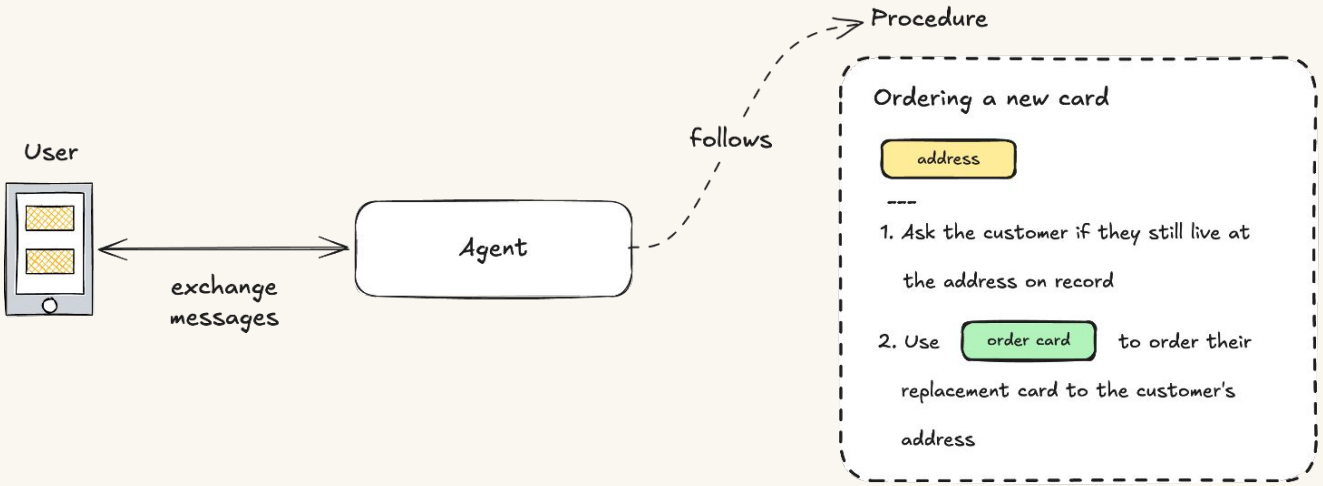
A high-level picture



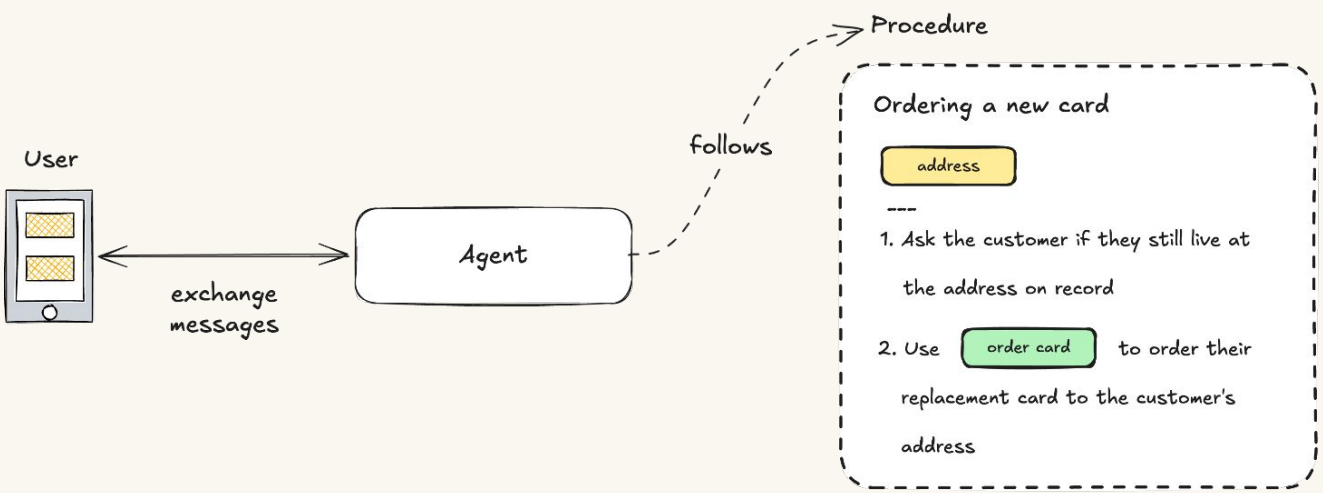
1. Make sure we're talking to the right person

2. Limit access to reading and writing the minimal data required

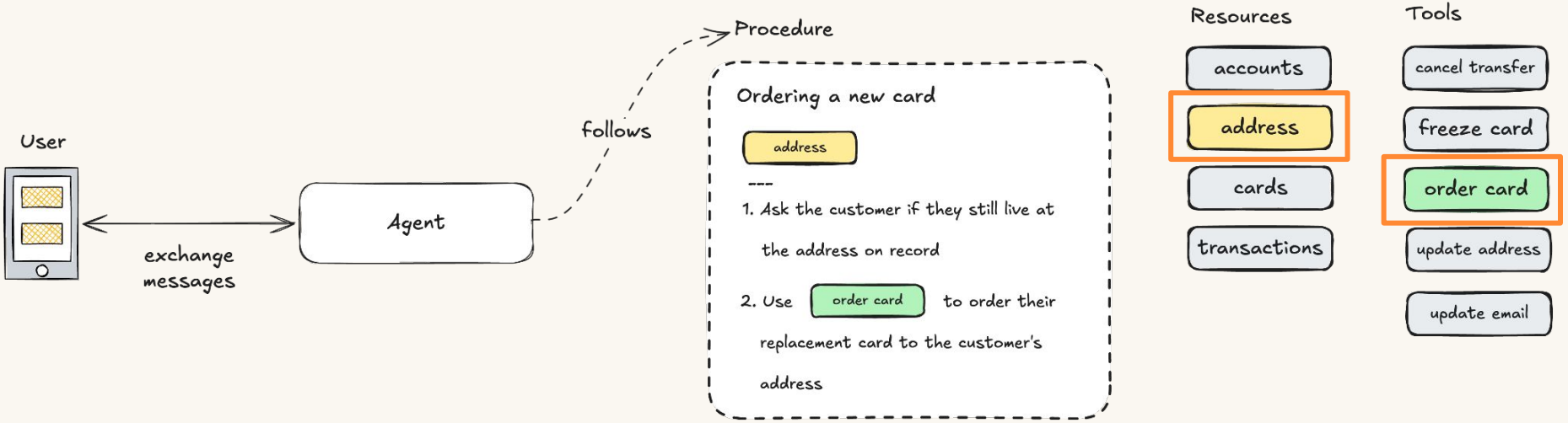
Limiting access



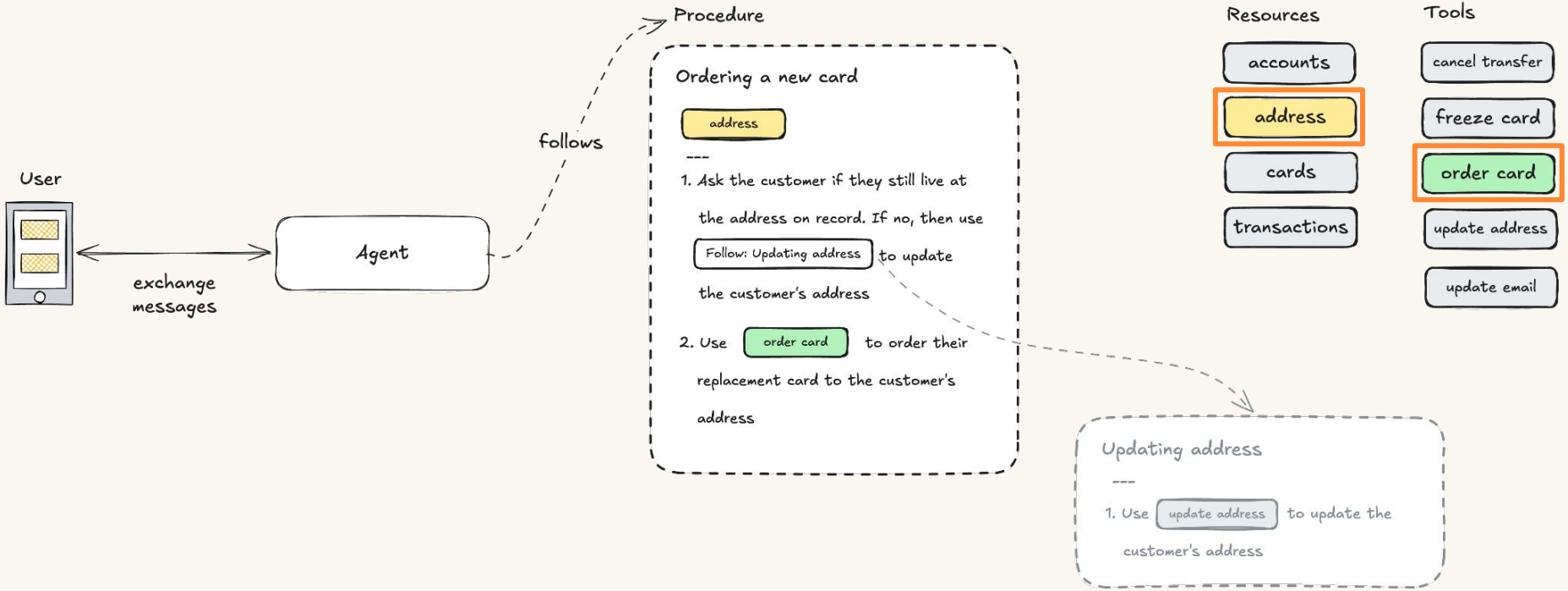
Limiting access



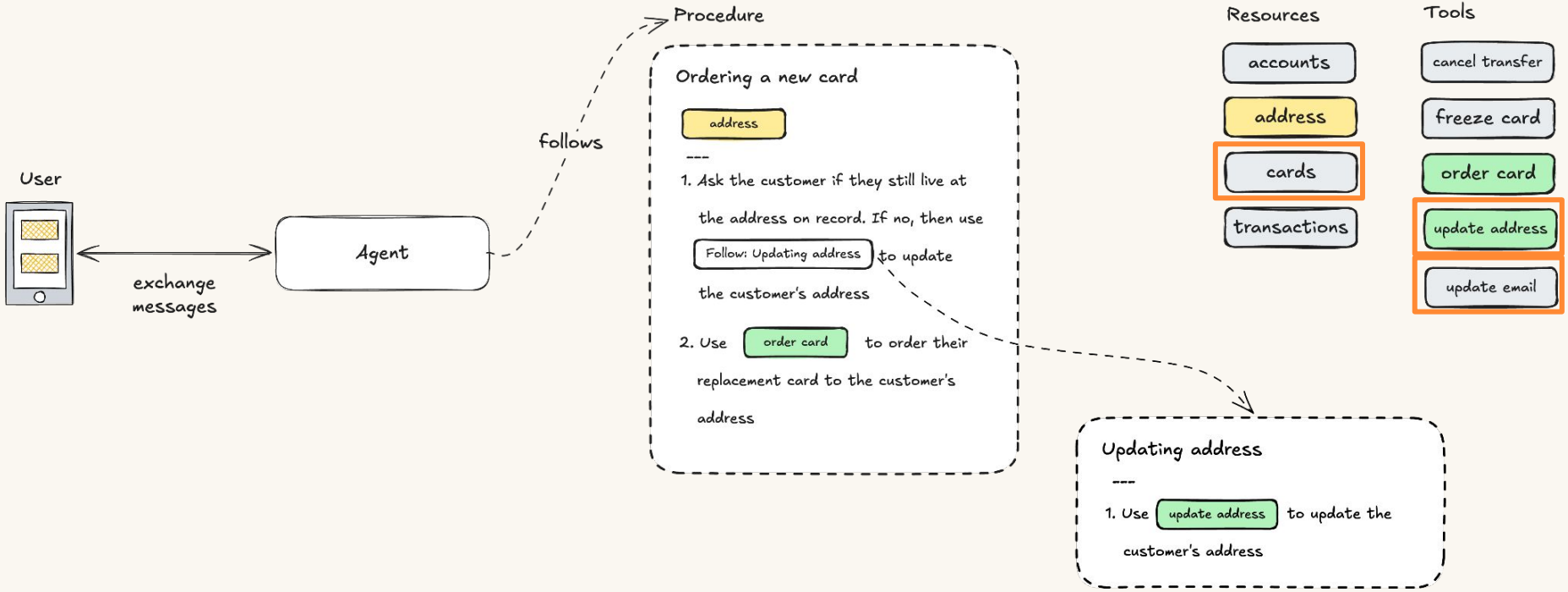
Limiting access



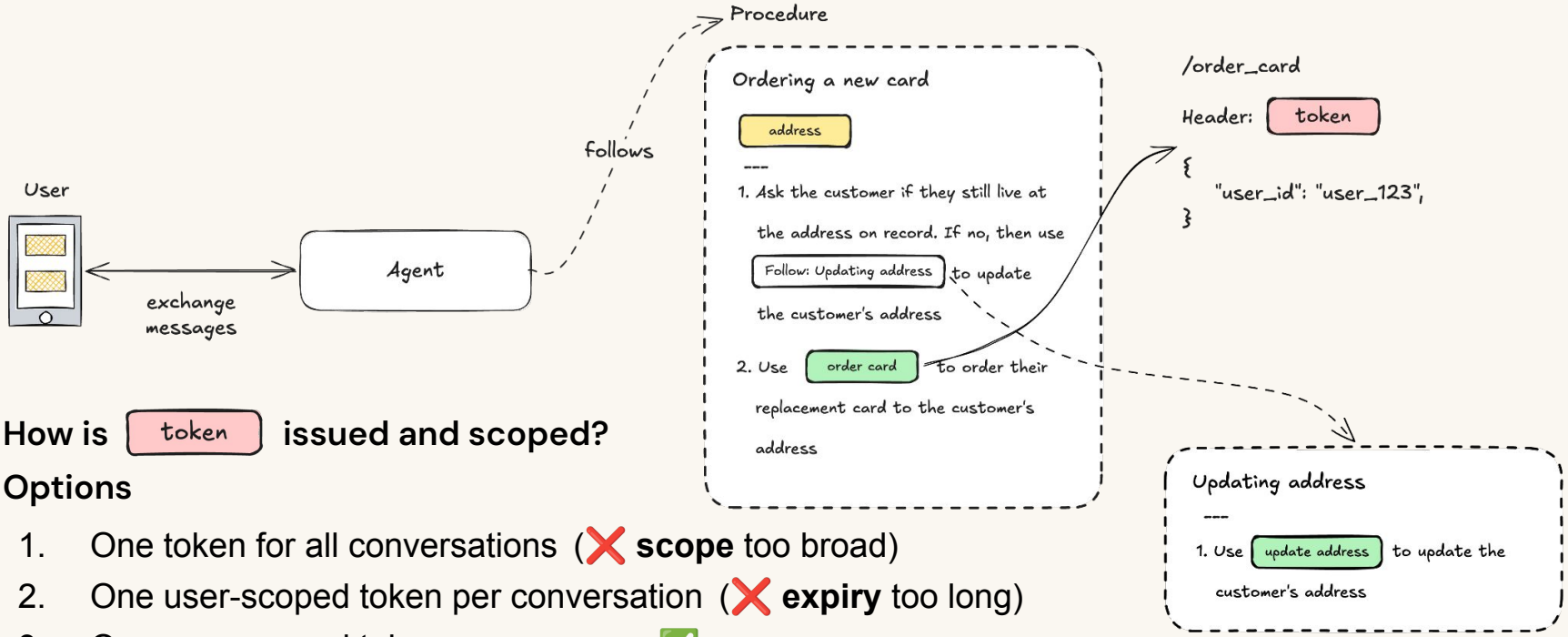
Limiting access



Limiting access



Limiting access

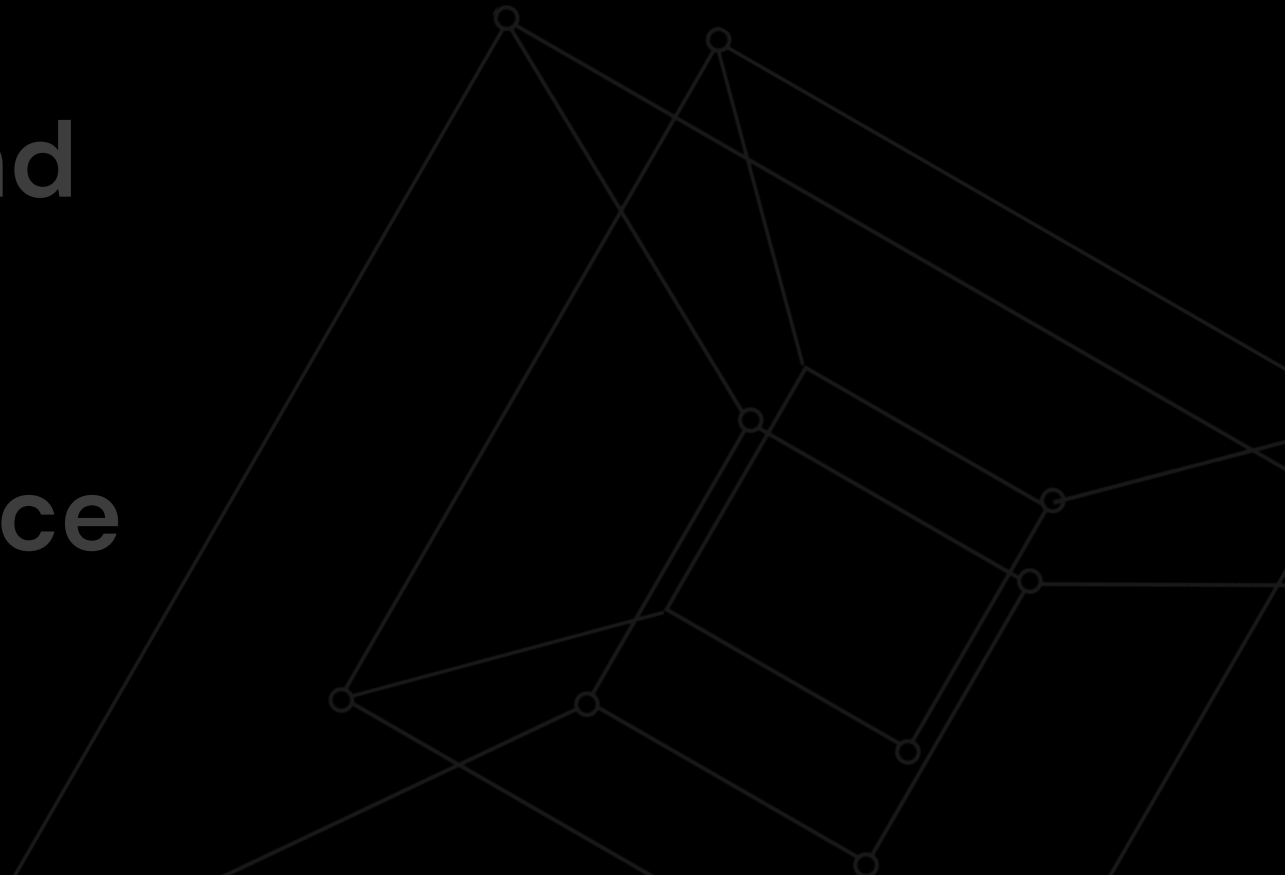


How is `token` **issued and scoped?**

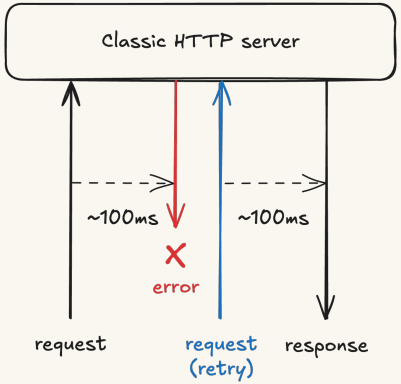
Options

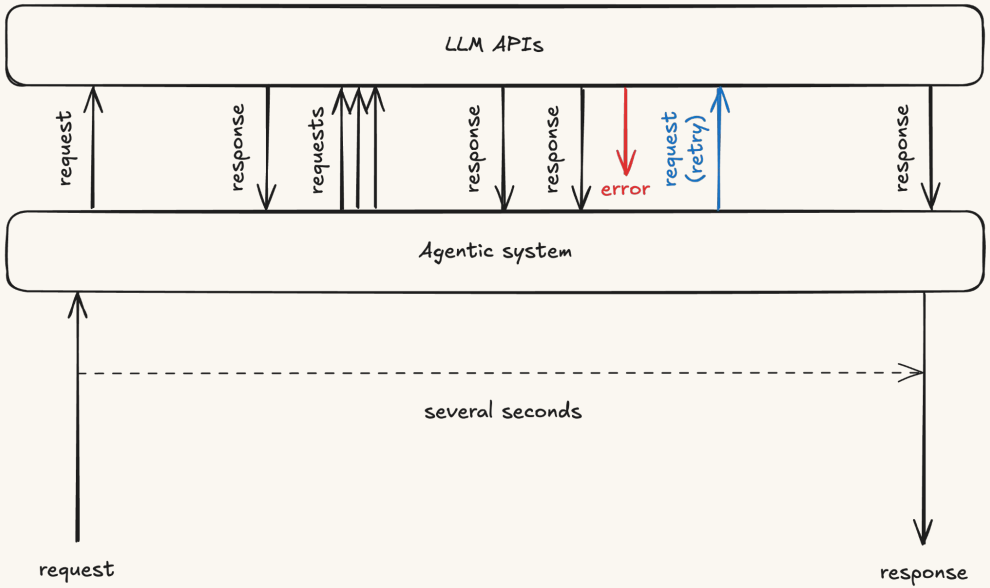
1. One token for all conversations (❌ **scope** too broad)
2. One user-scoped token per conversation (❌ **expiry** too long)
3. One user-scoped token per message ✅

Background
Security
Reliability
Performance



AI agents aren't like web servers





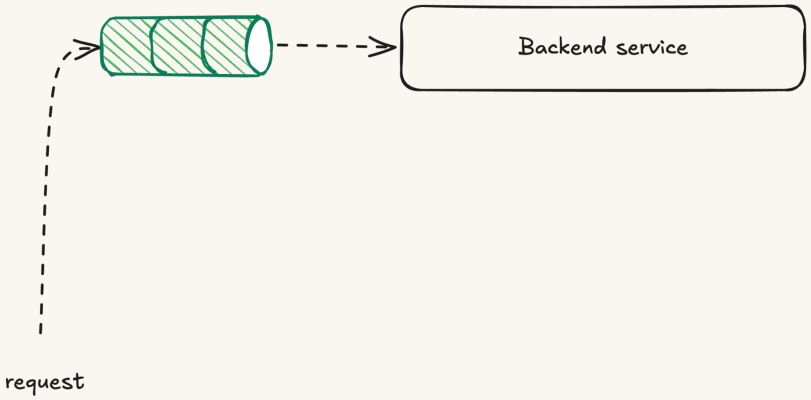
- Long-running
- Expensive
- Interruptible

Attempting durable execution

- Long-running
- Expensive
- Interruptible

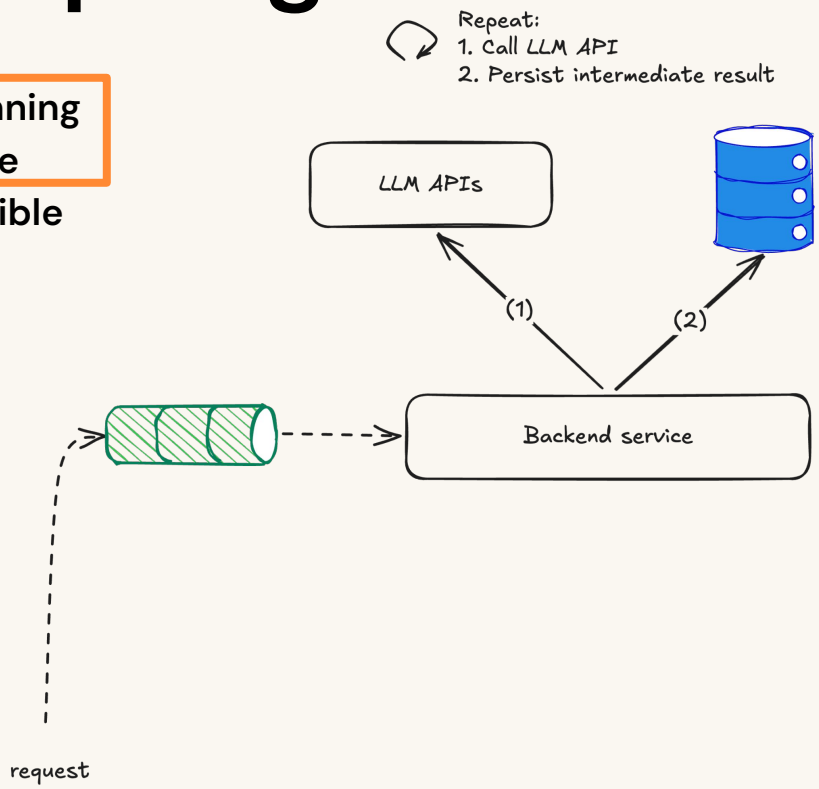
Attempting durable execution

- Long-running
- Expensive
- Interruptible



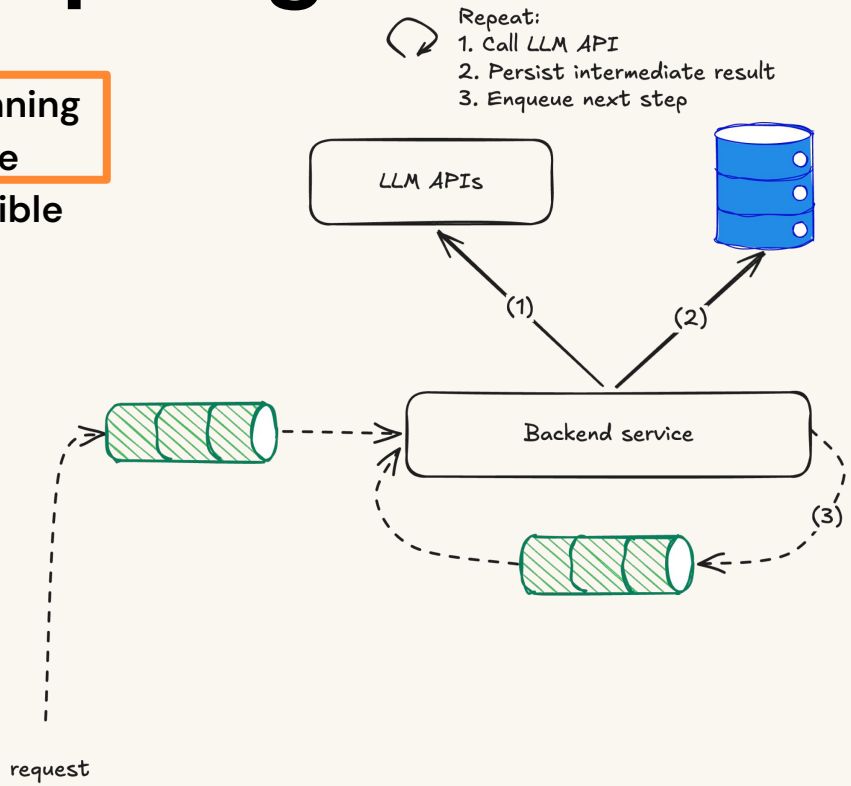
Attempting durable execution

- Long-running
- Expensive
- Interruptible



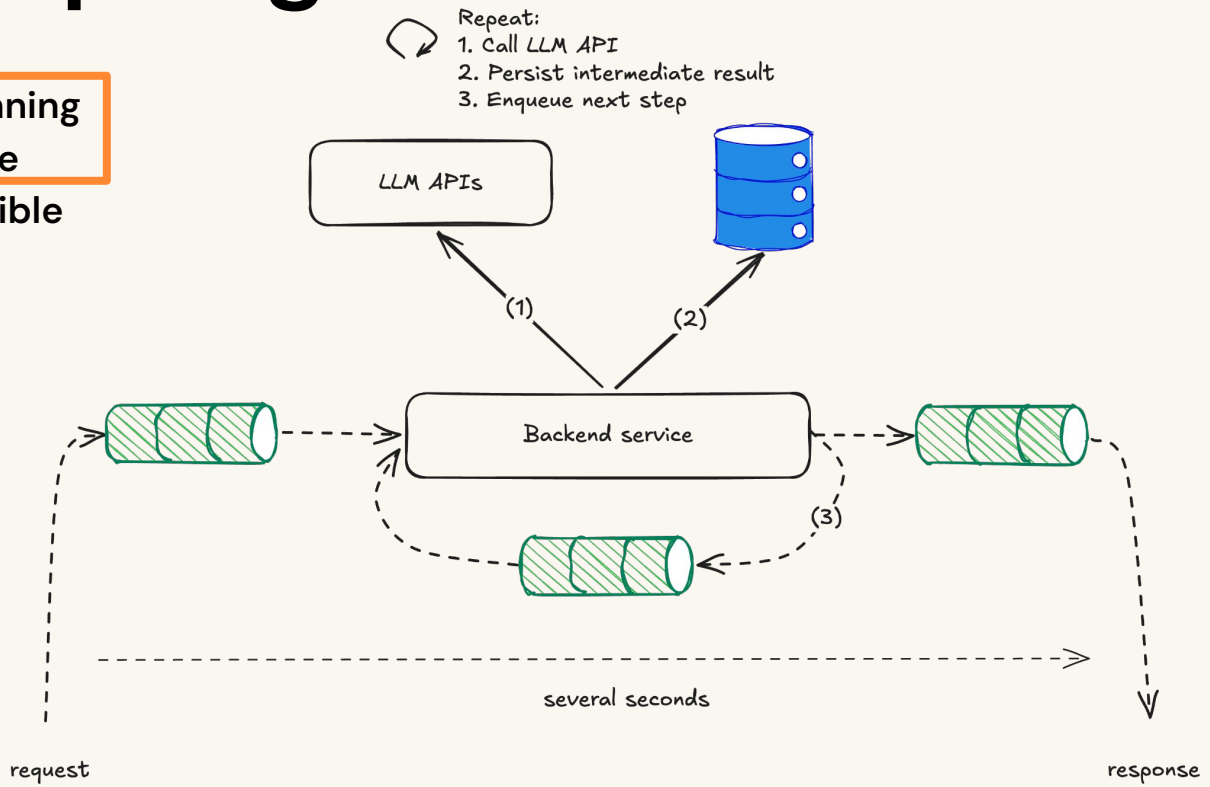
Attempting durable execution

- Long-running
- Expensive
- Interruptible



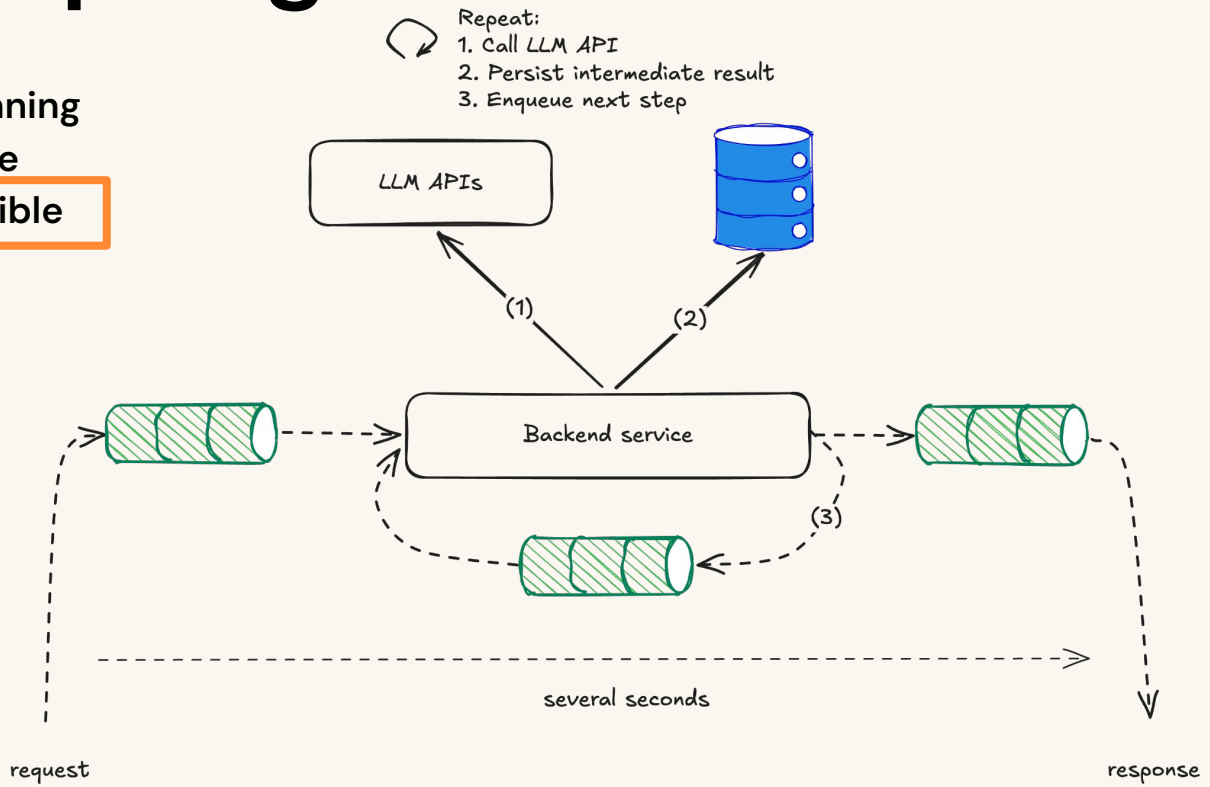
Attempting durable execution

- Long-running
- Expensive
- Interruptible



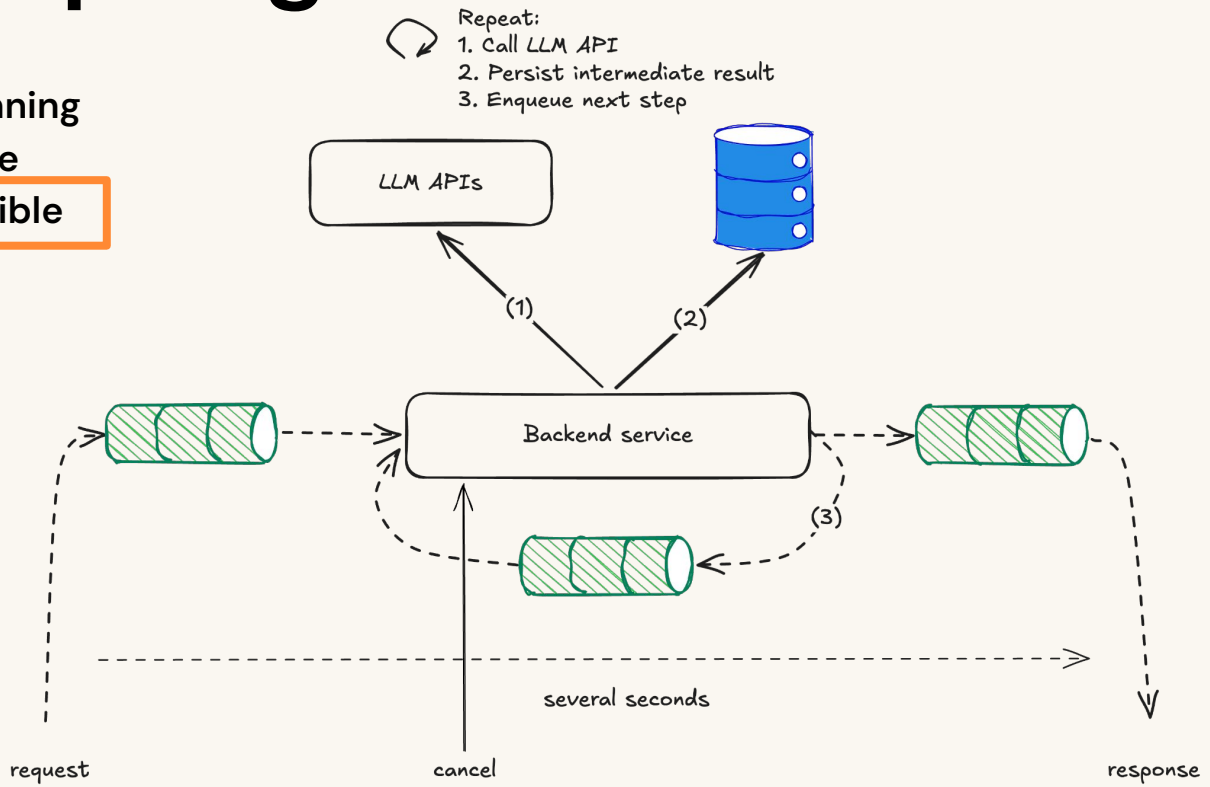
Attempting durable execution

- Long-running
- Expensive
- Interruptible



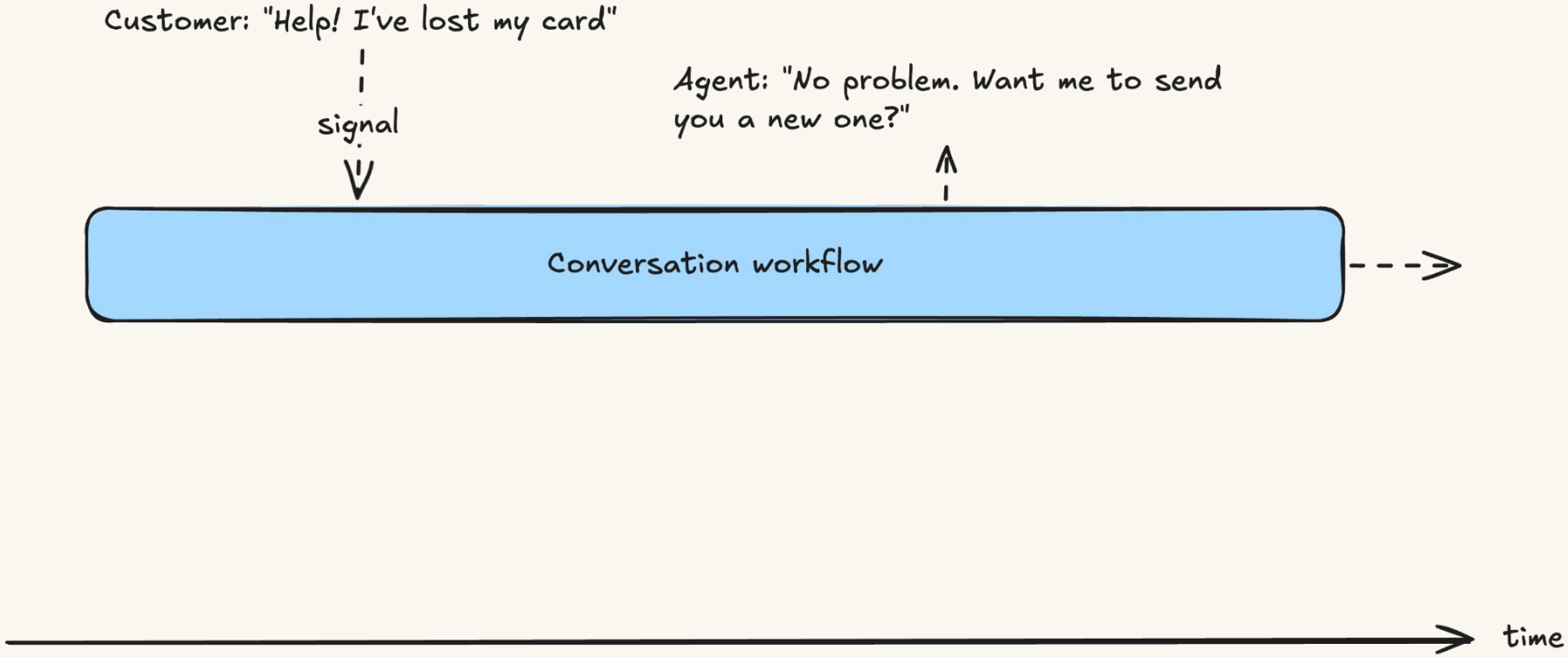
Attempting durable execution

- Long-running
- Expensive
- Interruptible



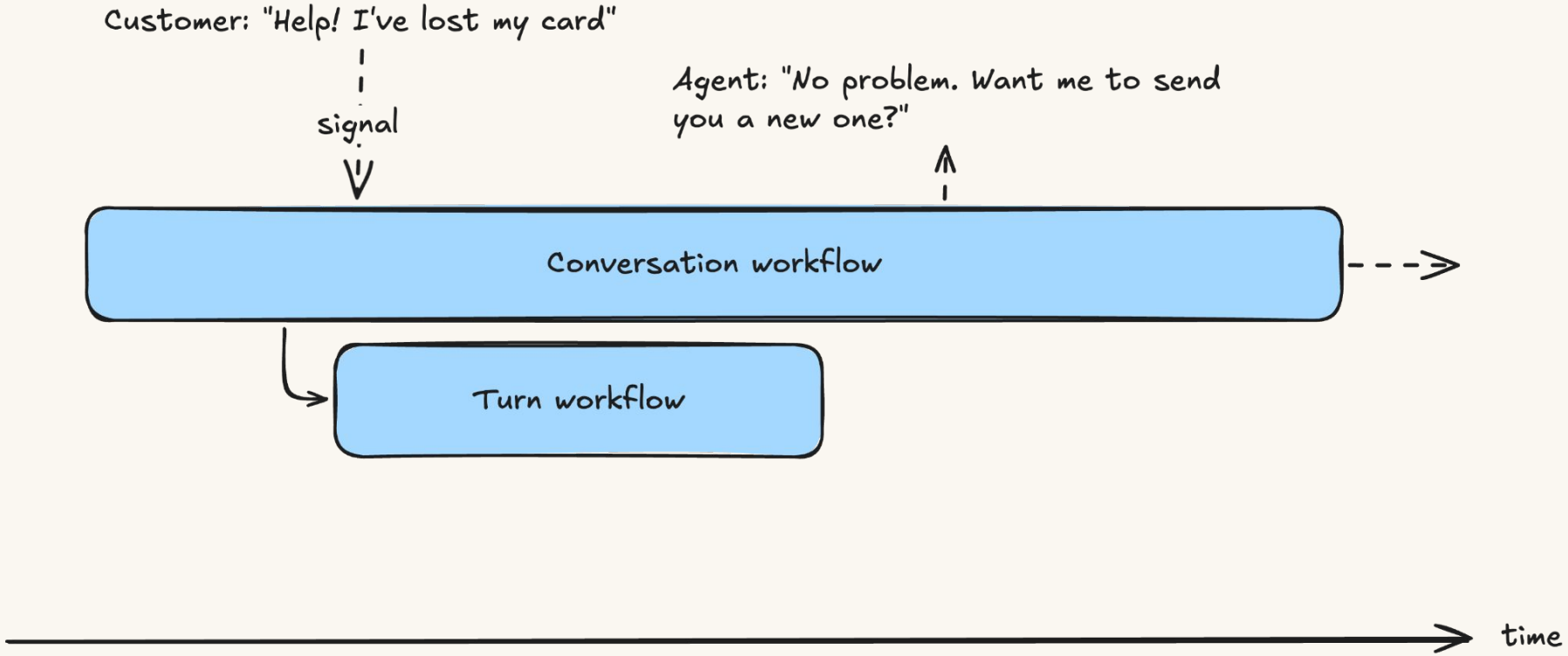
Durable execution

Temporal



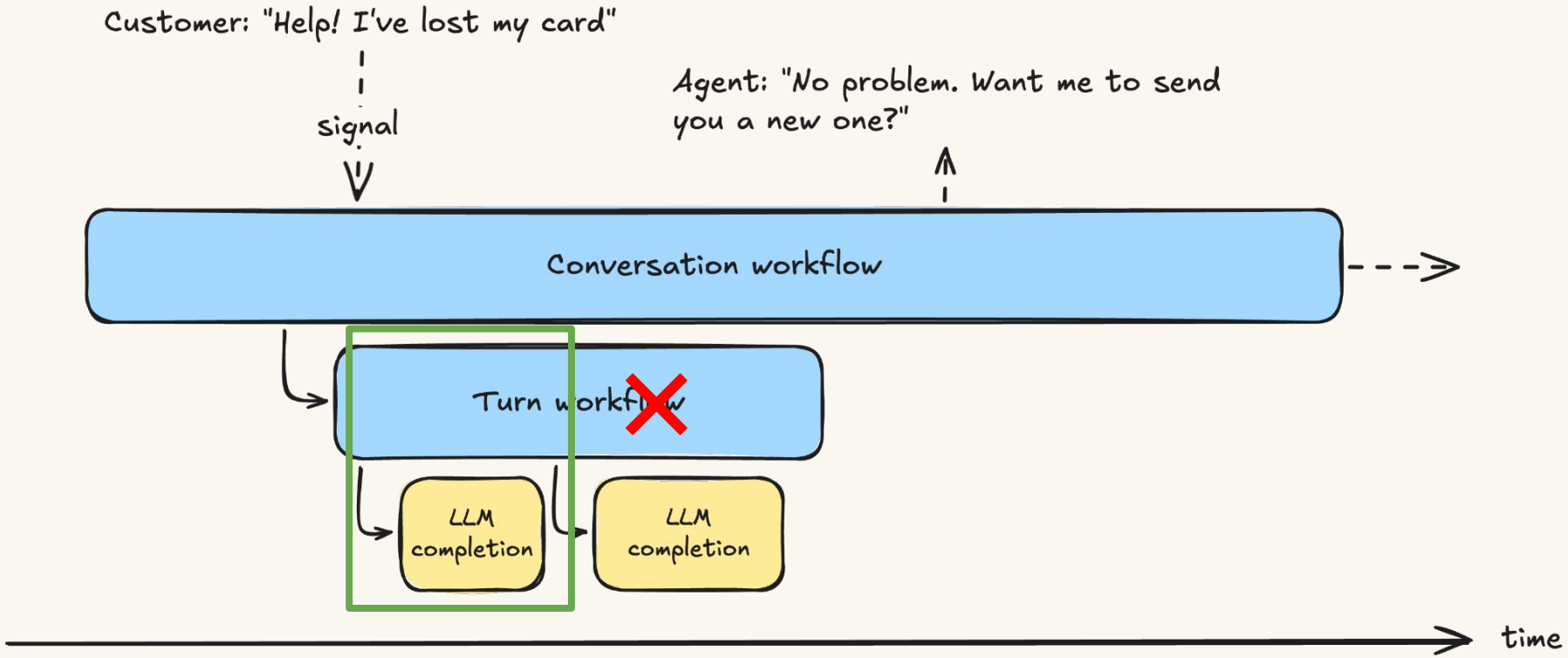
Durable execution

Temporal



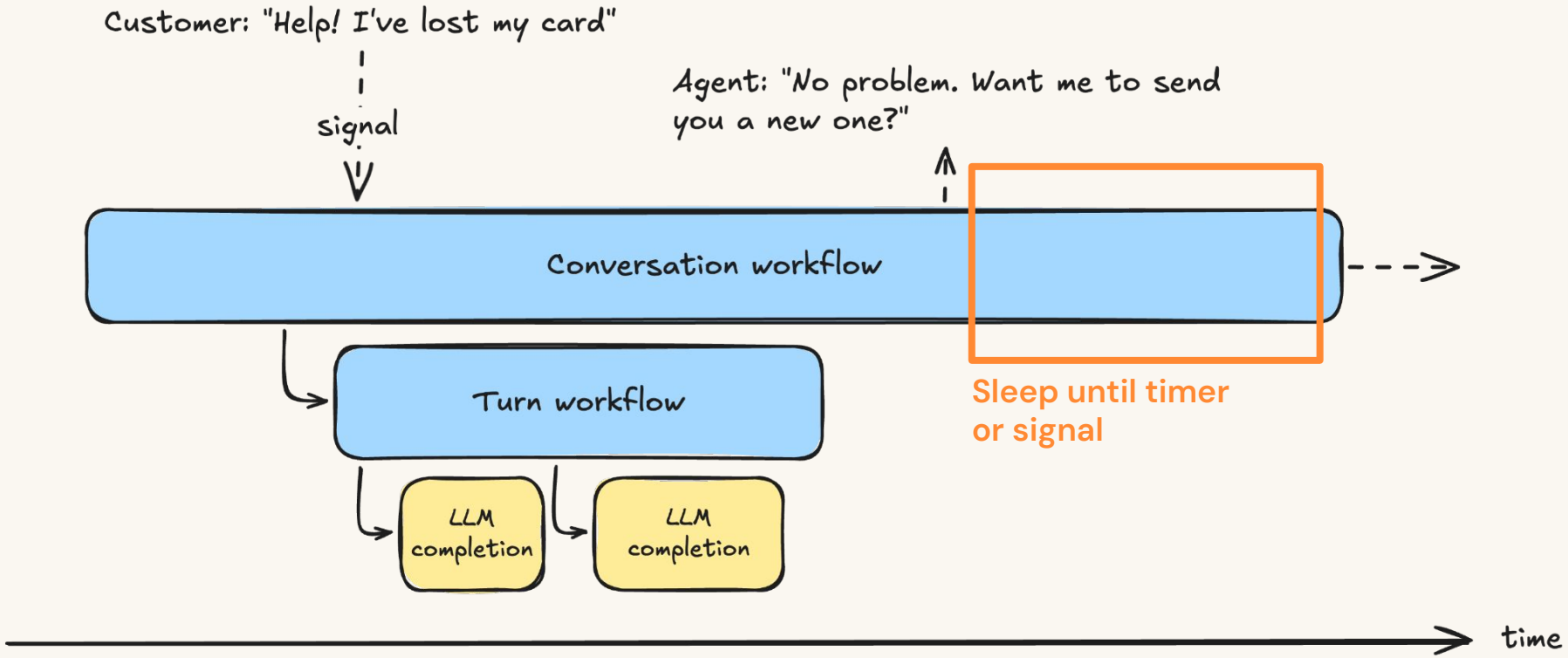
Durable execution

Temporal



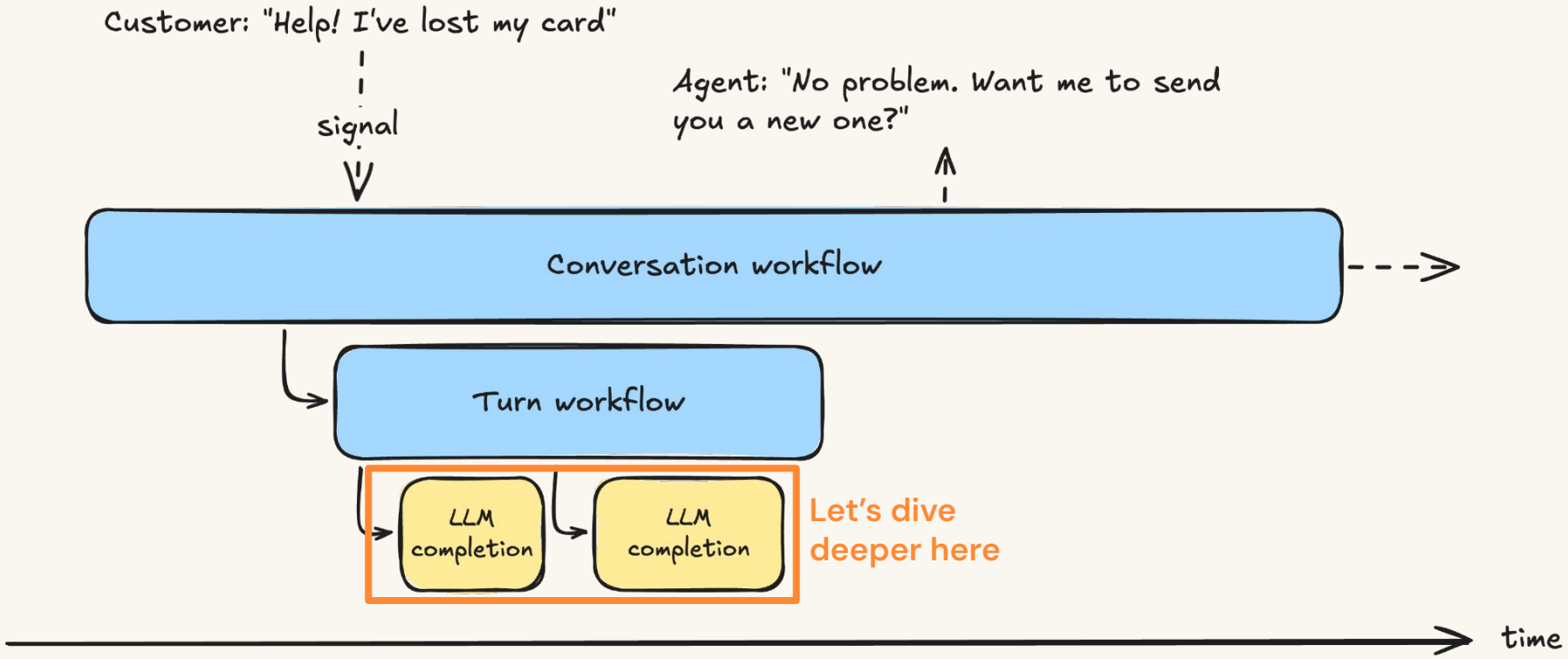
Durable execution

Temporal

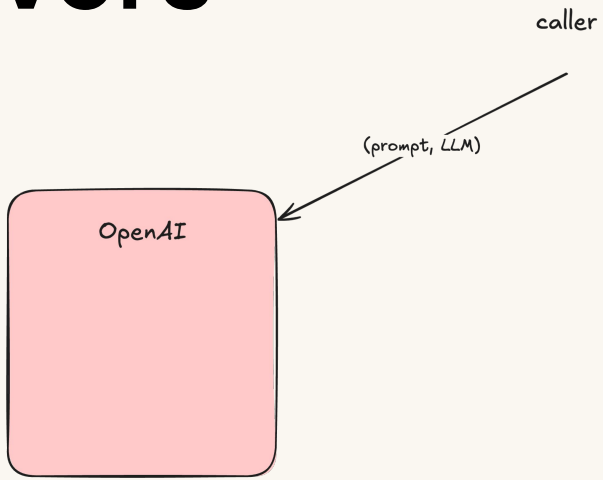


Durable execution

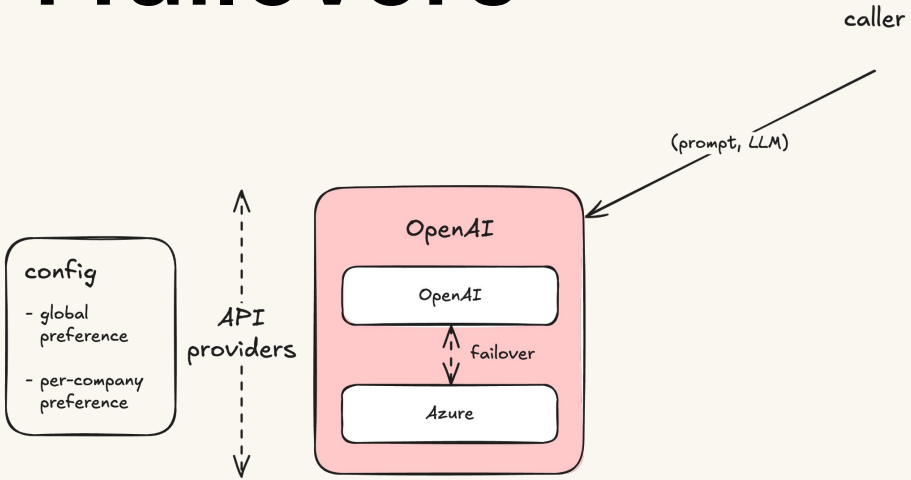
Temporal



LLM failovers



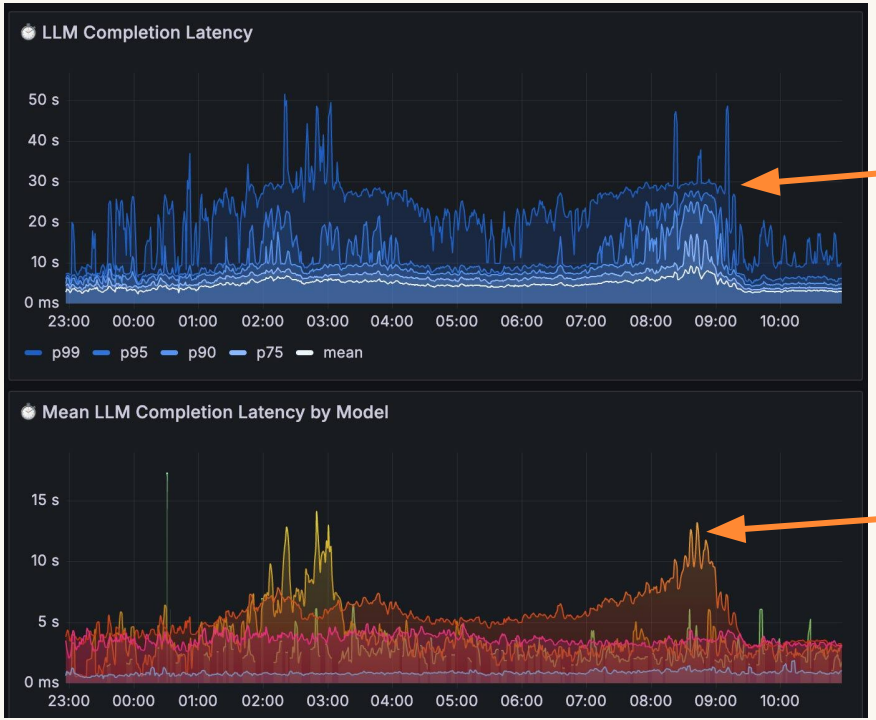
LLM failovers



When we fail over

- Errors (e.g. HTTP 5XXs)
- Rate limits
- High latency

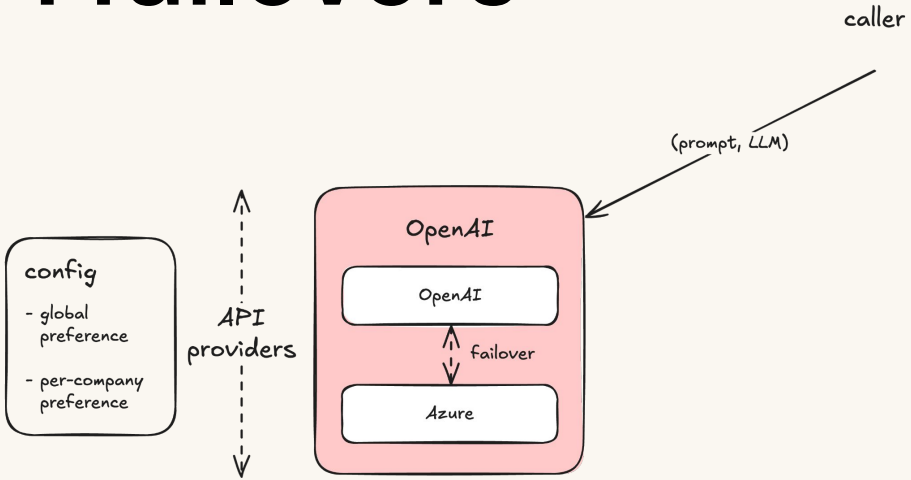
When to failover for latency?



P90 latency didn't change

But mean latency spiked

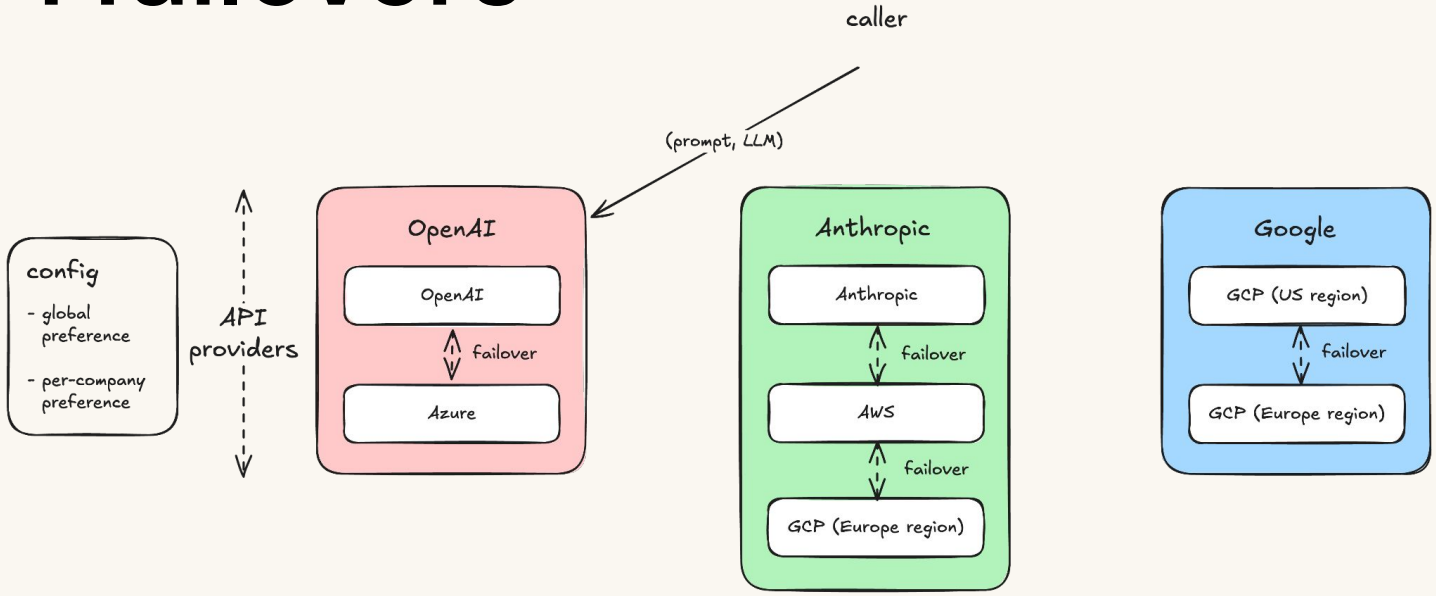
LLM failovers



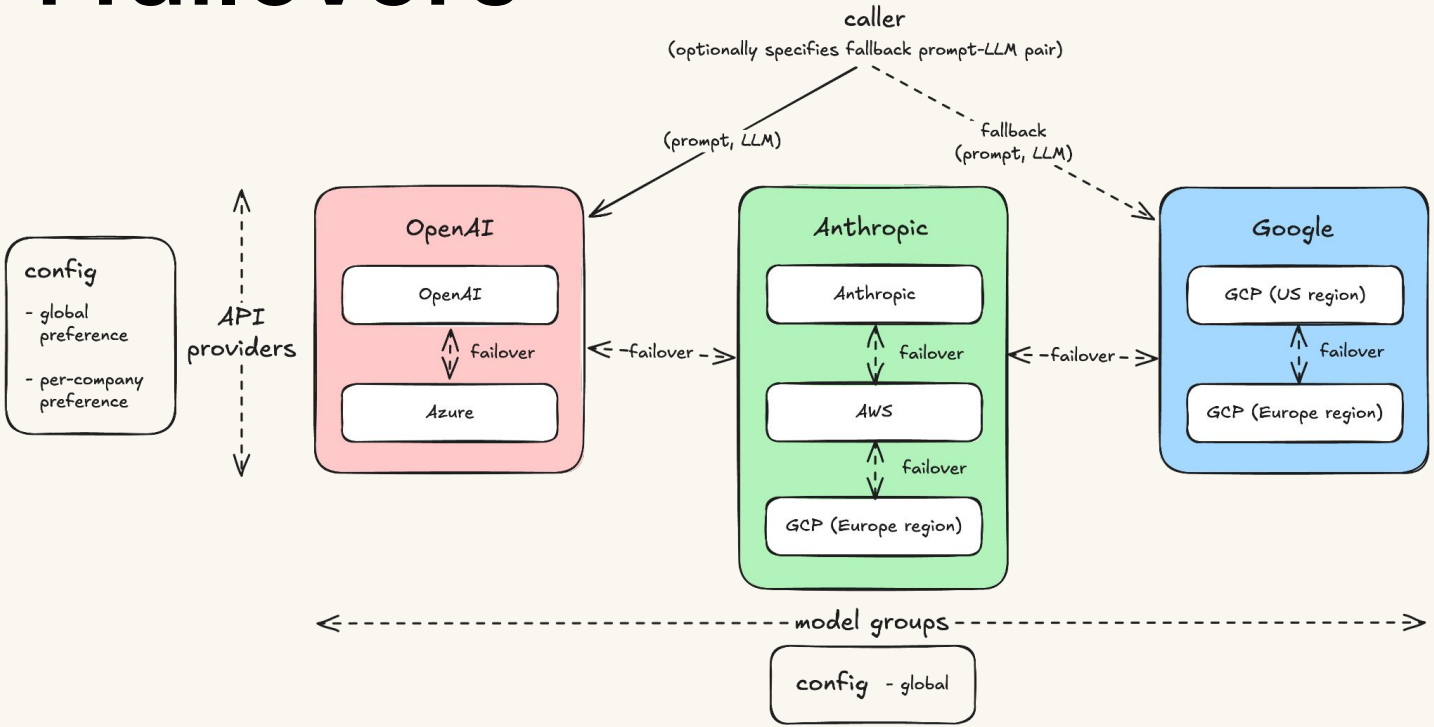
When we fail over

- Errors (e.g. HTTP 5XXs)
- Rate limits
- High latency

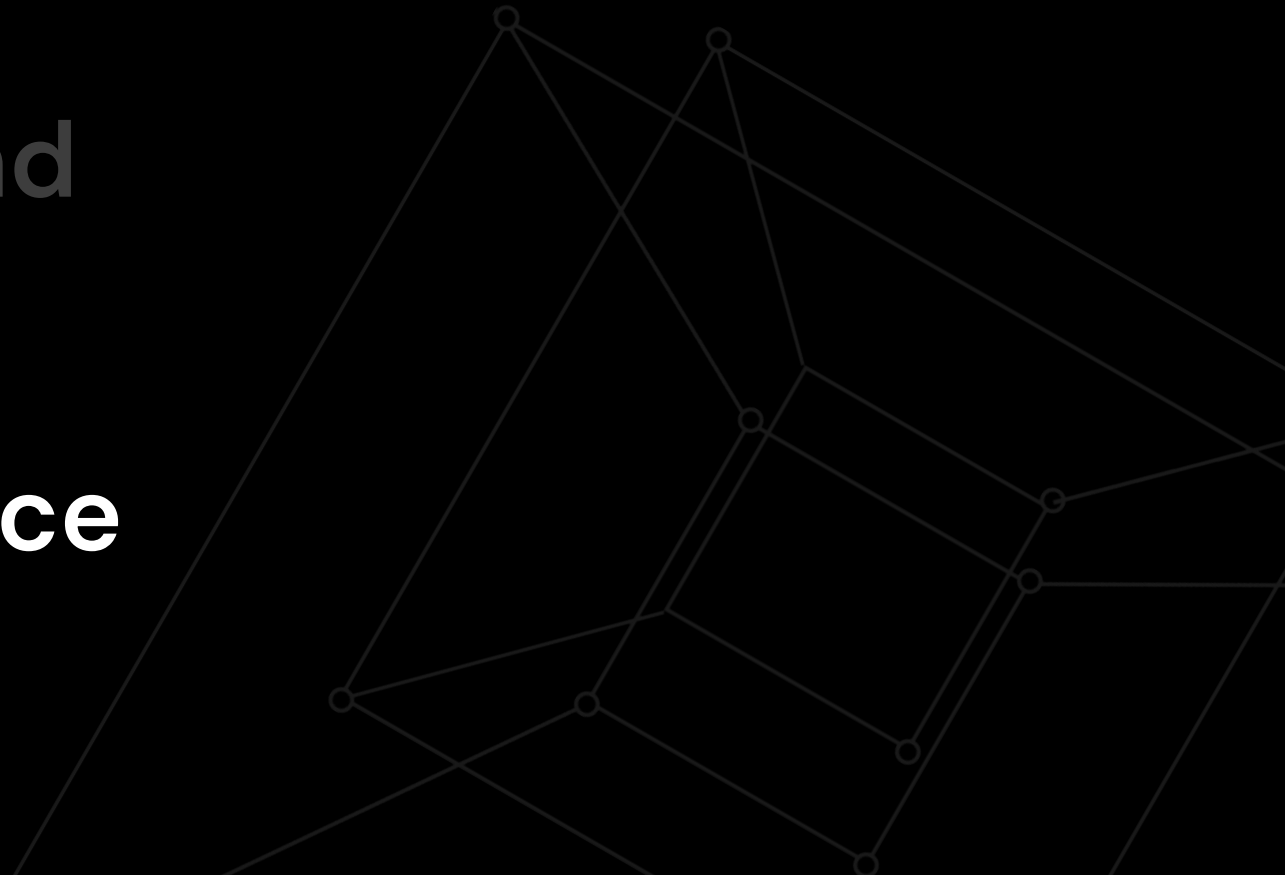
LLM failovers



LLM failovers



Background
Security
Reliability
Performance



HOT PATH

Performance is budget allocation

Quality

Enough evidence, correct procedure, valid tool arguments.

Latency

First audio, final answer, tool waits and timeout rate.

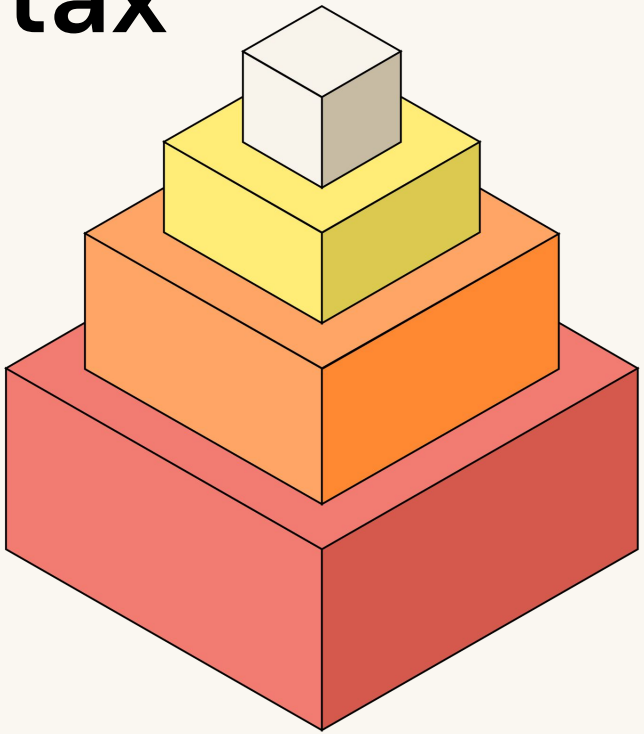
Cost

Model mix, rerank rate, guardrail triggers, retries.

Safety

Disclosure, IDV, vulnerability and post-tool checks.

Naive path: every turn pays every tax



Full policy retrieval

Broad resource search

Strong model

Expensive reasoning & latency

Guardrails

Every guardrail runs

Response

Answer or hold is delivered

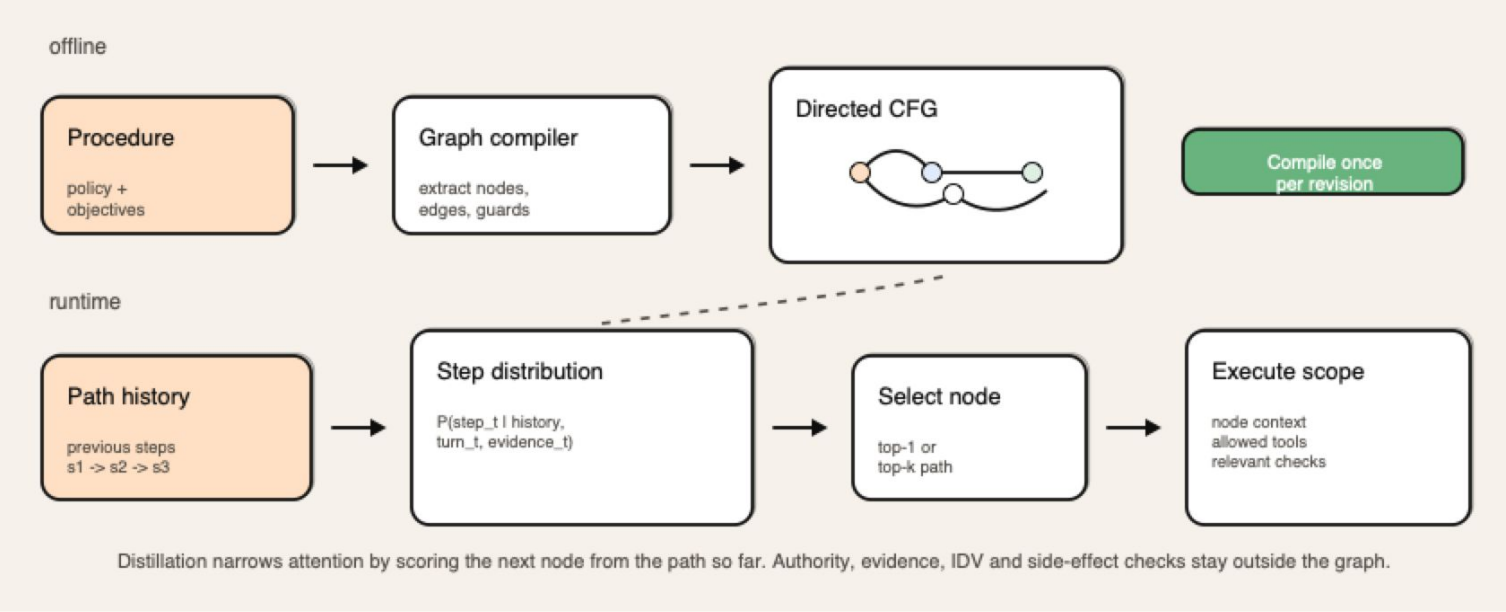
HOT PATH

Stop paying every tax

Naive block		Better move	Why it improves the hot path
Full policy retrieval	➡️	**Distill the active procedure**	Only carry the procedure chunks, tool bindings and forward links relevant to the predicted step.
Strong model every turn	➡️	**Toggle reasoning at runtime**	Use cheap classification for straight paths; pay for deeper reasoning only on ambiguity.
Every guardrail runs	➡️	**Pre-filter before guardrails**	Run expensive checks when the relevant risk is likely; fail open to full guardrails on uncertainty.

Same pattern each time: cheap routing first; expensive work only when it can change the outcome.

Distill the active procedure



Distillation narrows attention by scoring the next node from the path so far. Authority, evidence, IDV and side-effect checks stay outside the graph.

REASONING

Toggle reasoning at runtime

Straight path

Use no-reasoning / low-verbosity calls for narrow classification and rote policy turns.

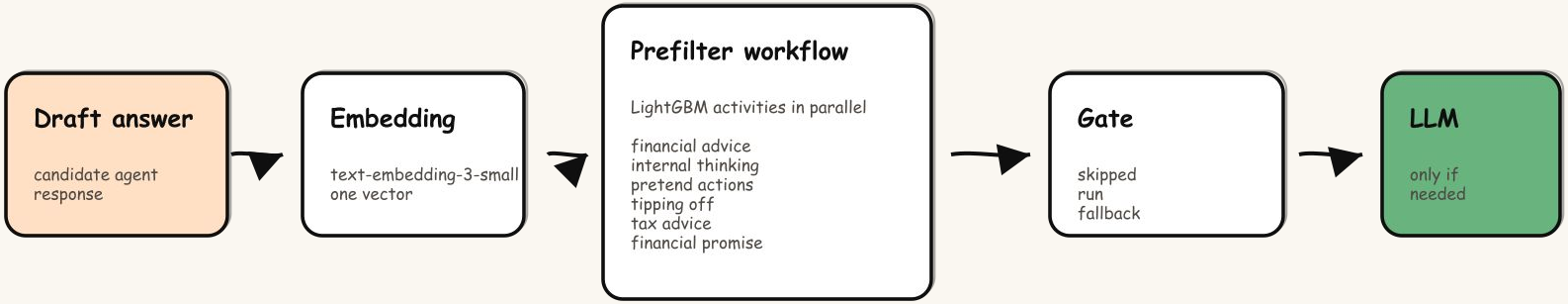
Step / limits classifier

Detect curveballs, evidence gaps or outside-procedure turns before paying for deeper reasoning.

Thinking path

Start stronger thinking execution only when ambiguity or customer-visible impact justifies it.

Prefilter gate, not replacement



High recall target

Optimize for catching non-compliance, not for replacing the judge.

Fail open

On timeout, missing prediction or model failure, run the full LLM guardrail.

Gradient Labs

Thank you

Danai & Devan

Gradient Labs

Did I mention we're hiring?

✨ gradient-labs.ai/careers ✨