

THE FIRST WORD IS NOT DONE



Streaming can show progress before generation completes.



It does not prove the answer is complete, valid, or correct.



Today: design timing, cancellation, and partial-result behavior.

Tell me a short story about a robot.



Model reply (streaming)

Once ...

First text appears here

Once upon a time ...

Once upon a time, there was a small robot ...

Once upon a time, there was a small robot who loved to explore the world.

Completed result here



Robot checks completed result.

CHOOSE THE USER EXPERIENCE

- **Live reply:** showing text early may reduce perceived waiting.
- **Background summary:** a finished, validated artifact may be more useful.
- **Choose streaming from the task**, not because the API supports it.

LIVE REPLY (STREAMING)

Show text early to reduce perceived waiting.



Explain how photosynthesis works.



Photosynthesis is the process by which plants convert light ...



It takes place in chloroplasts, where light energy is converted to chemical energy ...



This allows plants to make glucose from carbon dioxide and water.



Stream complete

BACKGROUND SUMMARY (JOB)

Produce a finished, validated artifact that may be more useful.



Summarize this document and give me key takeaways.



Job submitted

We'll notify you when it's ready.



Processing in background

Analyzing, summarizing, and validating...



Summary ready

Here are your key takeaways.

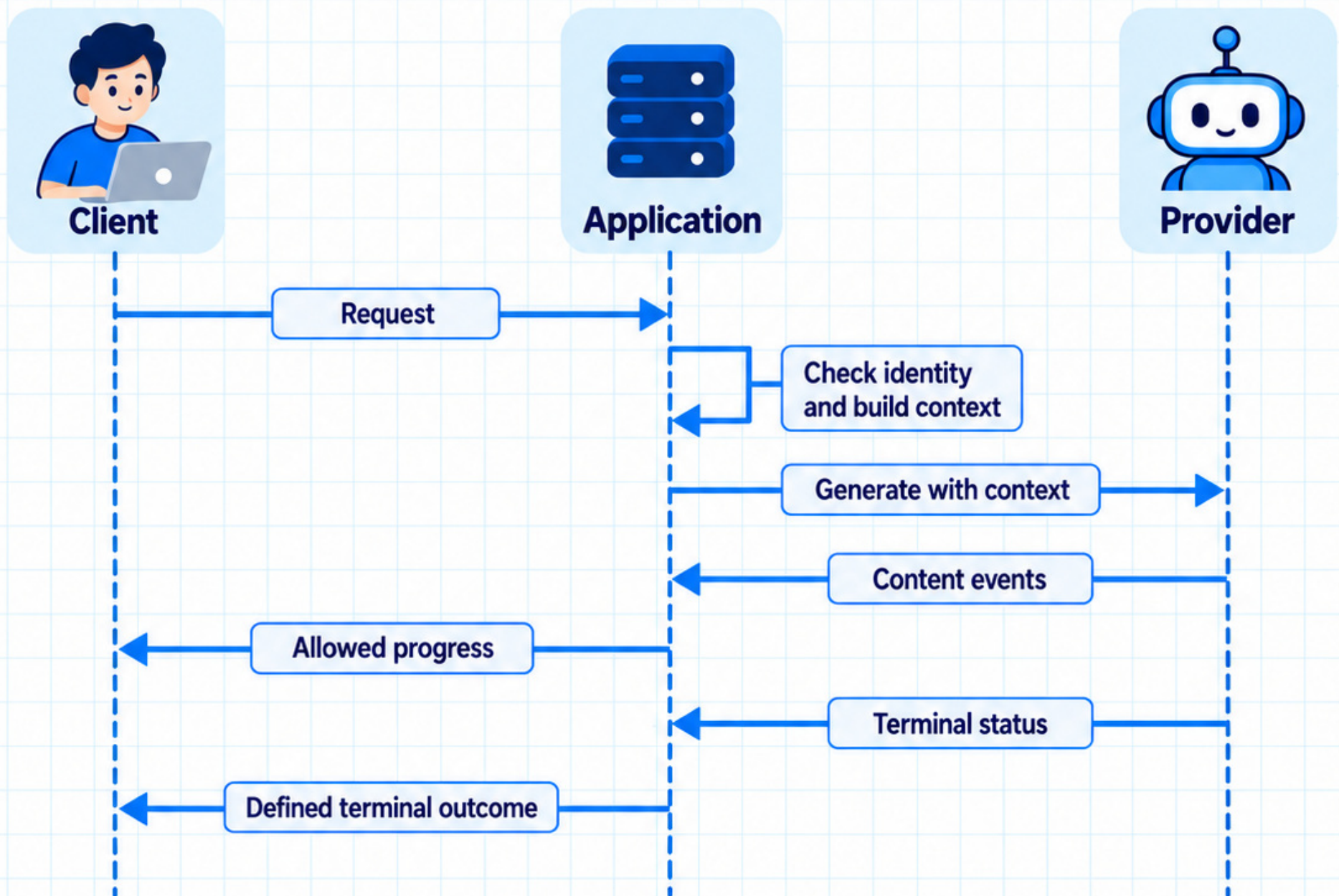
[View summary](#)

DRAW THE FULL SEQUENCE

Client sends a request.
Application checks
identity and builds
context.

Provider emits events.
Application handles
and forwards allowed
content.

Client renders progress,
then a defined
terminal state.

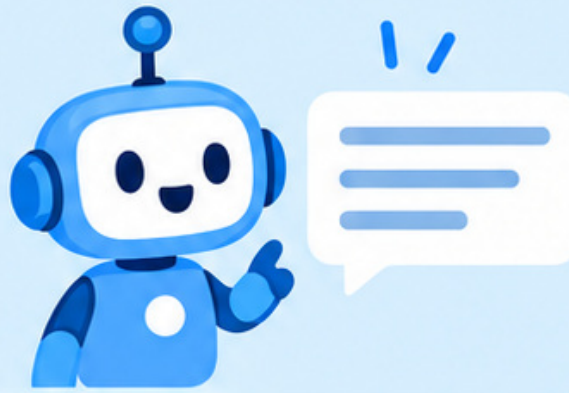


MEASURE THREE MOMENTS

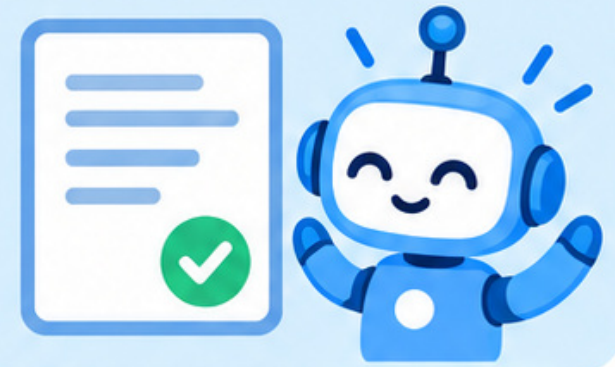
Request start:
the user begins waiting.



First useful text:
the user sees meaningful content.



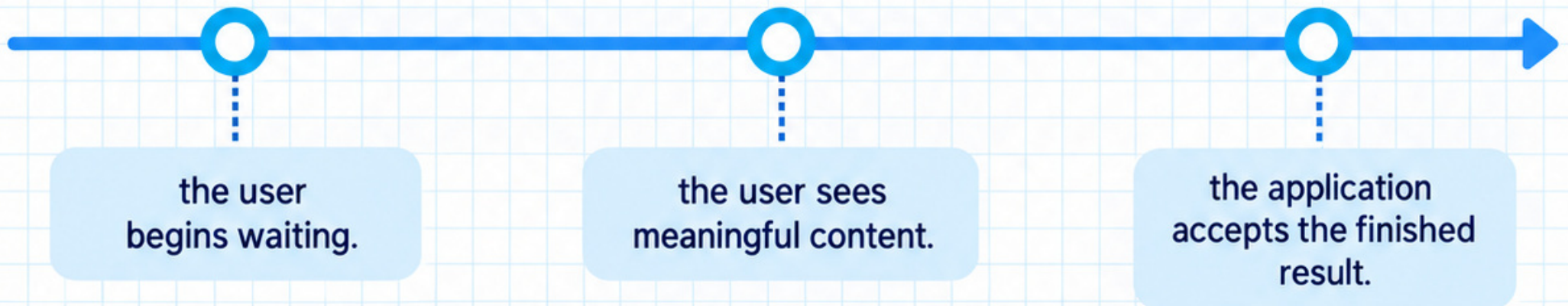
Usable completion:
the application accepts the finished result.



Request start

First useful text

Usable completion



Measure each at a clearly named boundary.

A TOY TIMING EXAMPLE



Illustrative client measurements:

Request starts at 0 s. First useful text arrives at 1 s.
Usable completion arrives at 5 s.

First-text time = 1 s. Total completion time = 5 s.

Request starts

First useful text arrives

Usable completion arrives



Partial output between 1 s and 5 s



0 s
(toy)

1 s
(toy)

5 s
(toy)

First-text time = 1 s

Total completion time = 5 s



STREAMING DOES NOT REMOVE WORK



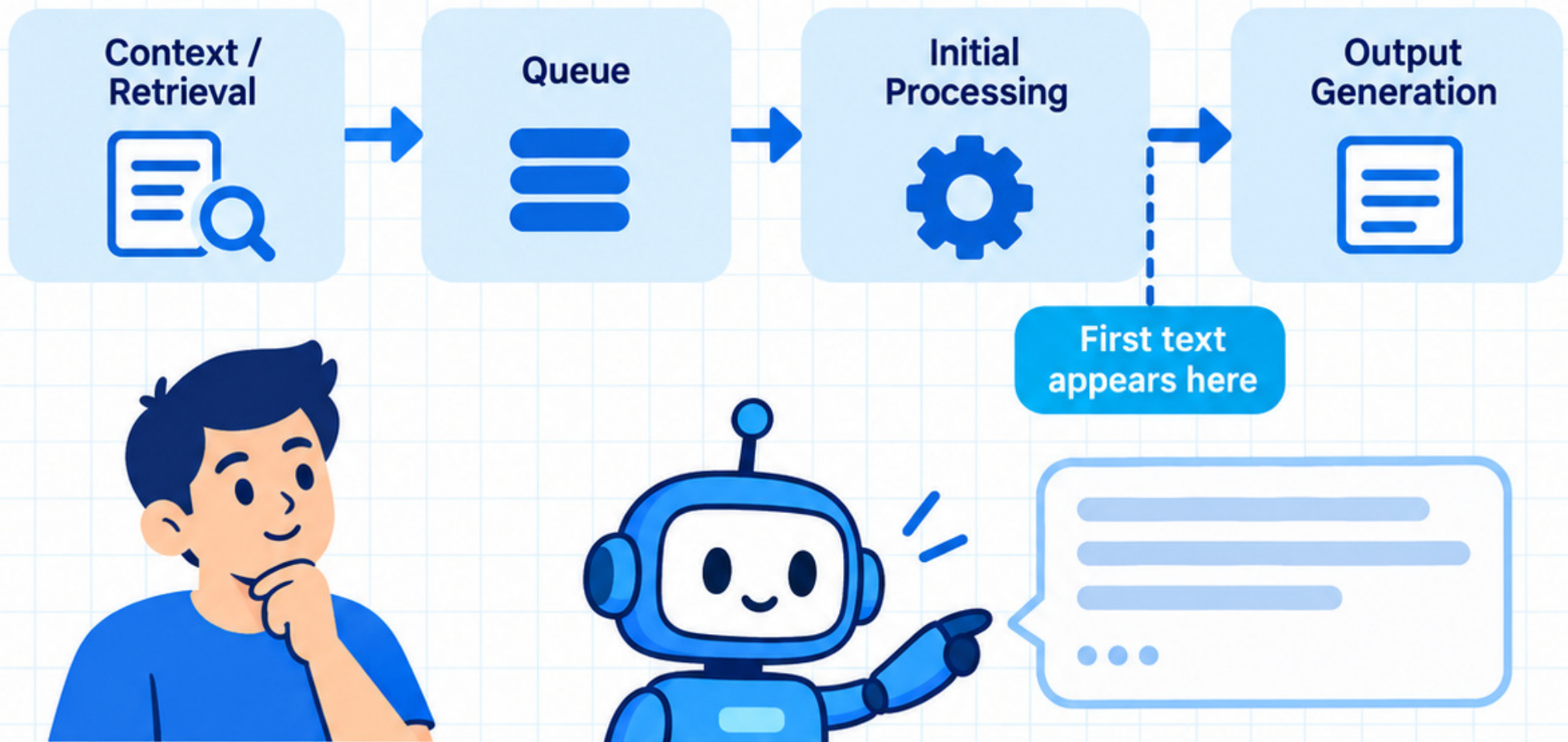
Retrieval, queueing, and initial model processing can happen before text appears.



A faster first chunk may leave total completion unchanged.

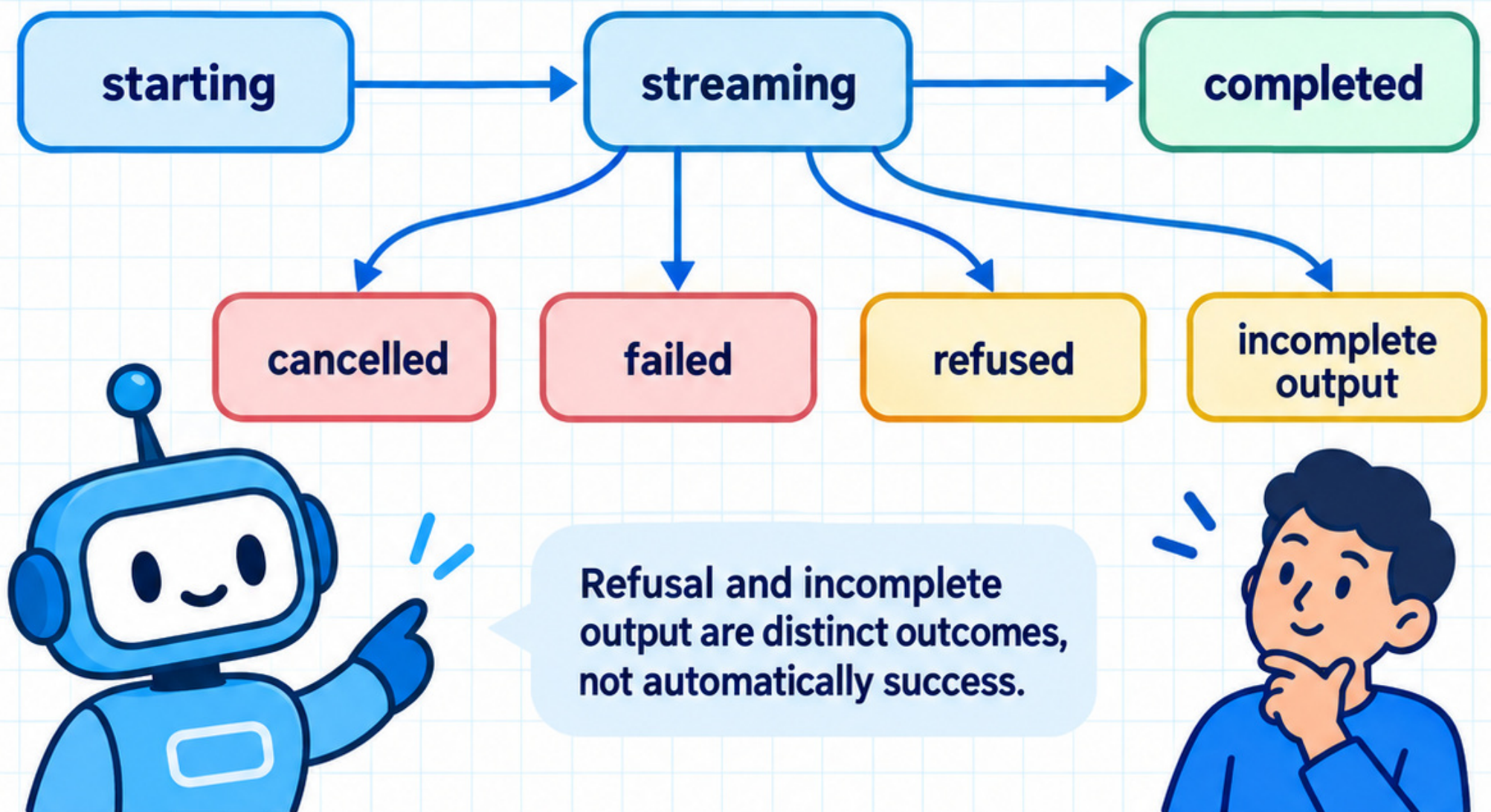


Measure the entire path under representative load.



DEFINE STREAM STATES

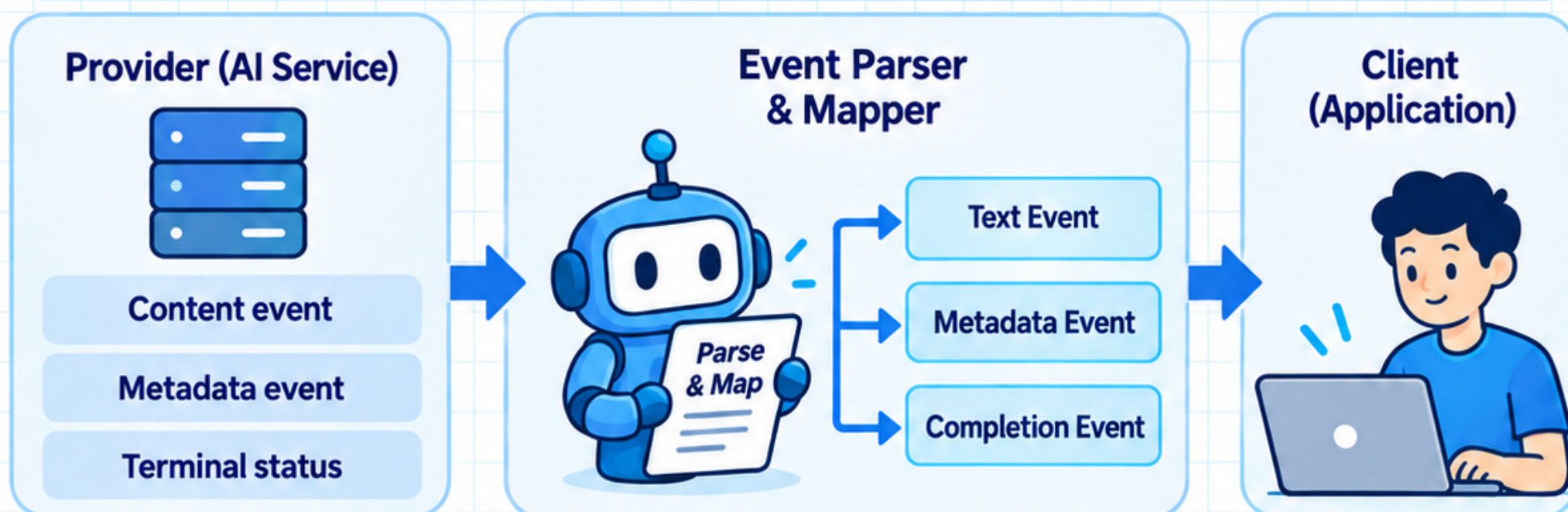
- Use explicit states: starting, streaming, completed, cancelled, and failed.
- Track refusal and incomplete output as distinct outcomes.
- Do not equate a closed connection with successful completion.



FORWARD EVENTS DELIBERATELY

Provider events may contain text deltas, metadata, or terminal status.
Parse the documented event format and map it to your application protocol.

Illustrative event flow; provider formats vary.



A network chunk is not necessarily a complete token, sentence, or JSON value.

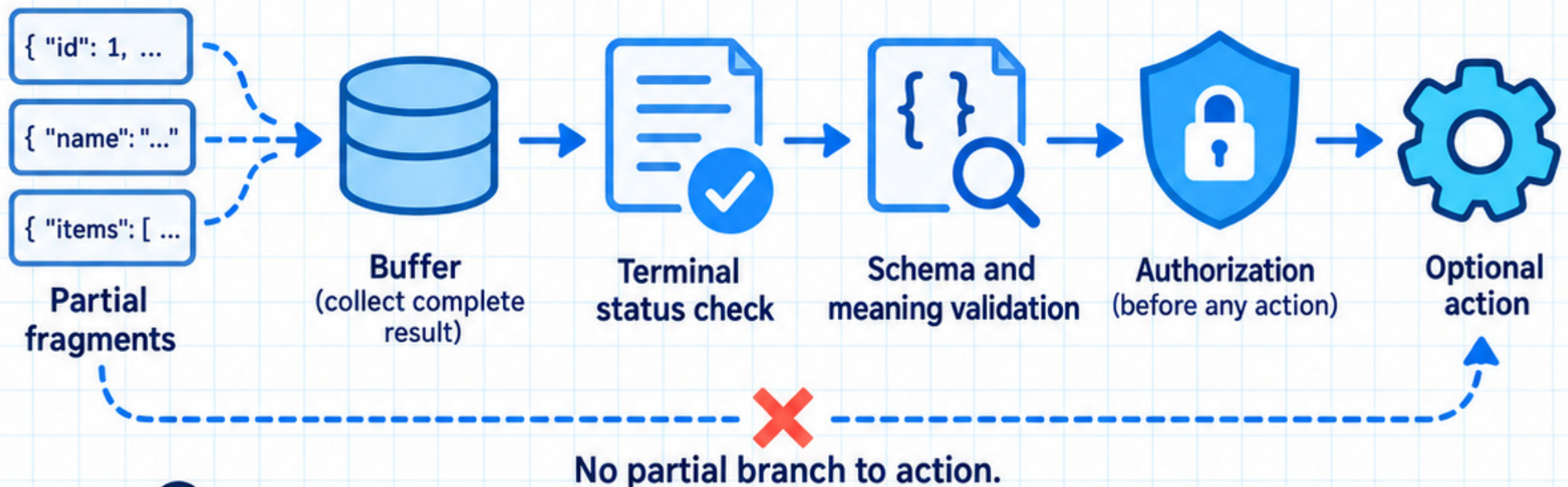


VALIDATE AFTER COMPLETION

For structured output, collect the complete result and check its schema and meaning.

Partial JSON is not a valid business result.

Tool arguments need application authorization before any action.



LABEL PARTIAL TEXT HONESTLY

- Display unfinished text as in progress.
- If generation fails, mark it incomplete and explain the recovery option.
- Never silently present a cut-off answer as a completed response.

Generating a response...



Here are some key considerations for designing an AI system at scale...



In progress



The answer is still being generated.



Generation failed.



Here are some key considerations for designing an AI syst...



Incomplete

The response was cut off due to a generation error. You can retry to get a complete answer.

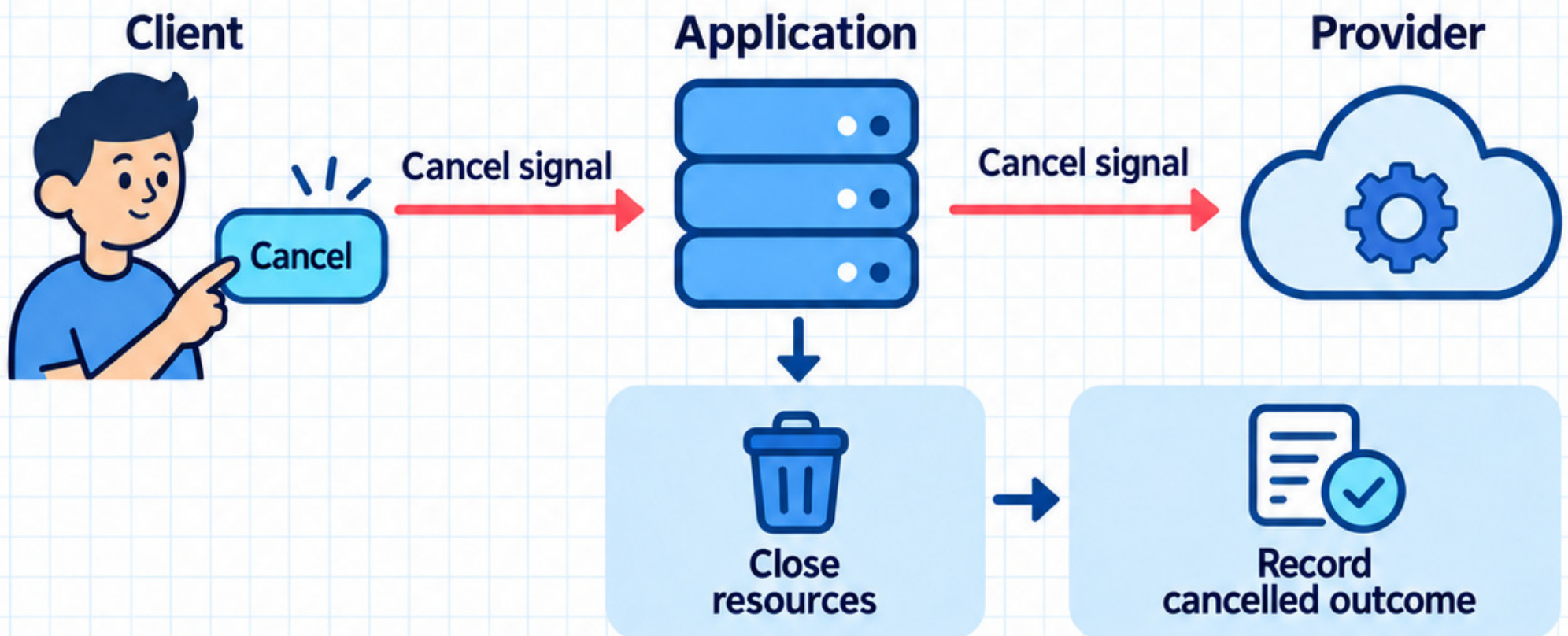
Retry

CANCELLATION MUST TRAVEL

When the user cancels, stop forwarding content and request upstream cancellation where supported.

Close resources and record the outcome.

Some work or charges may already have occurred.



No guarantee of immediate stop or zero bill.

SET GENERATION LIMITS

Define output size, overall deadline, idle timeout, and allowed retries. Use limits supported by the chosen provider and enforce application deadlines.

LIMIT POLICIES

Output size



Define the maximum size of the generated output.

Overall deadline



Set the maximum time allowed for the entire request.

Idle timeout



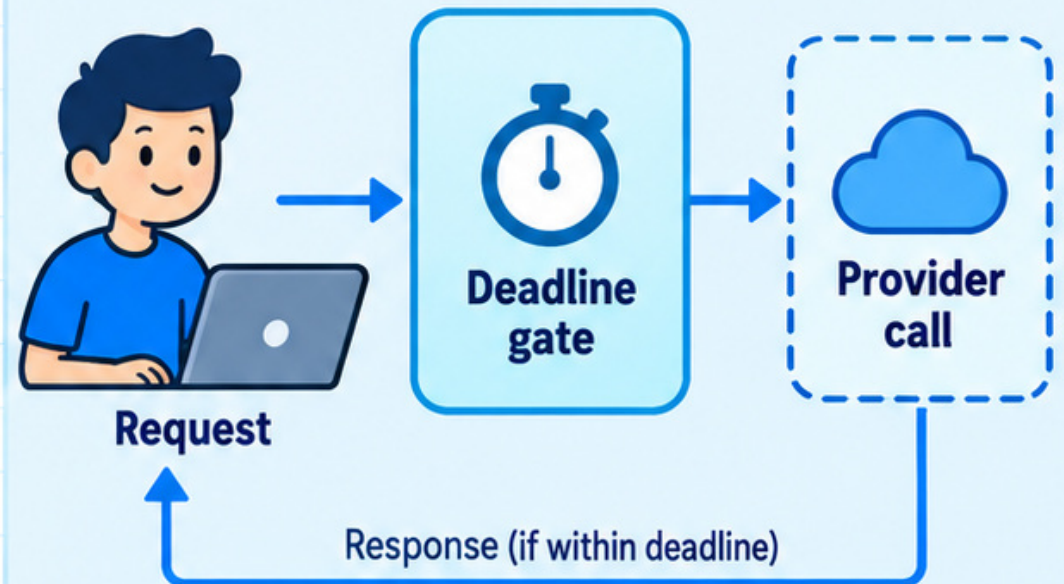
Stop if there is no activity for a period of time.

Allowed retries



Limit how many times a request can be retried.

ENFORCE DEADLINE AROUND PROVIDER CALL



A retry can repeat cost and content; keep attempts identifiable.

IDENTIFY ATTEMPTS

Attempt 1



Attempt 2



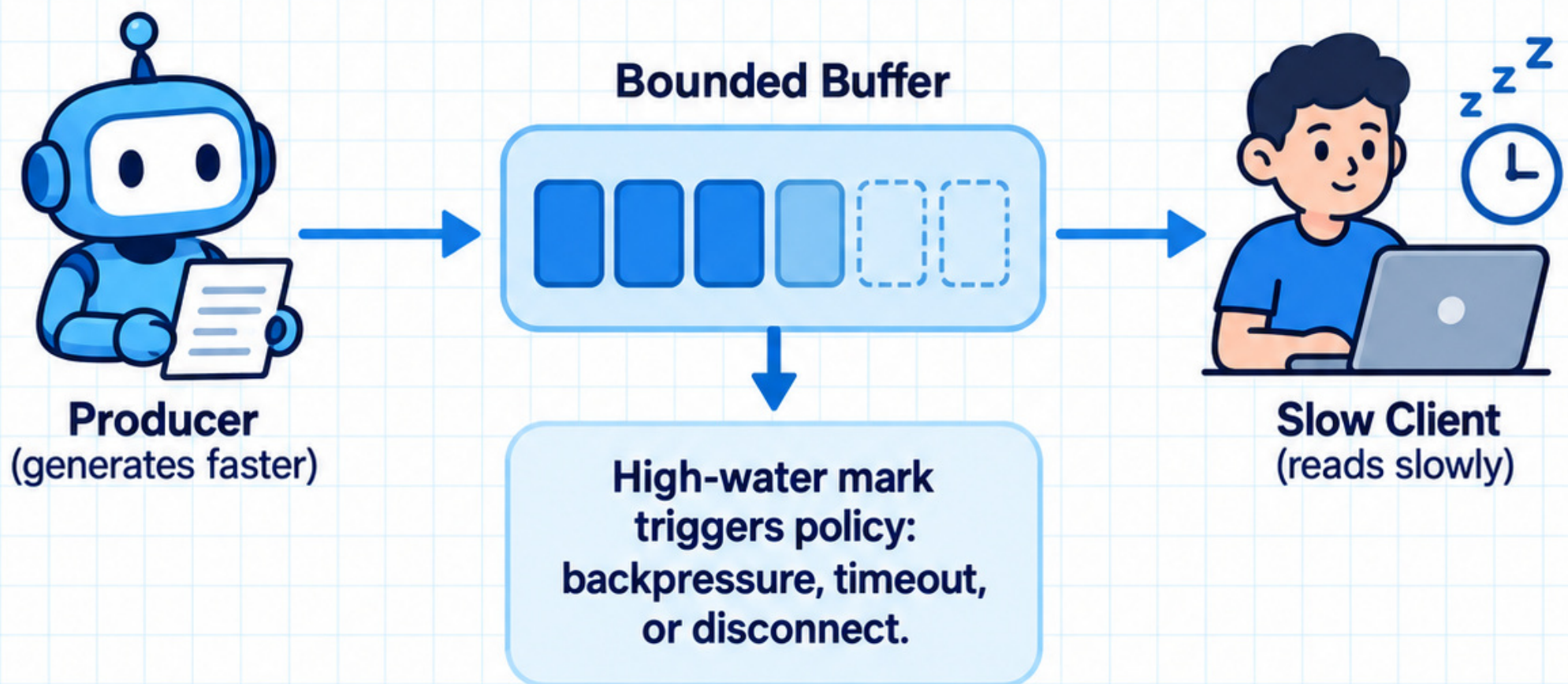
Attempt 3



HANDLE A SLOW CLIENT

A producer may generate faster than the client reads. Bound buffers and define backpressure, timeout, or disconnect behavior.

An unlimited queue can consume memory without improving the answer.



KEEP PERMISSION CHECKS EARLY



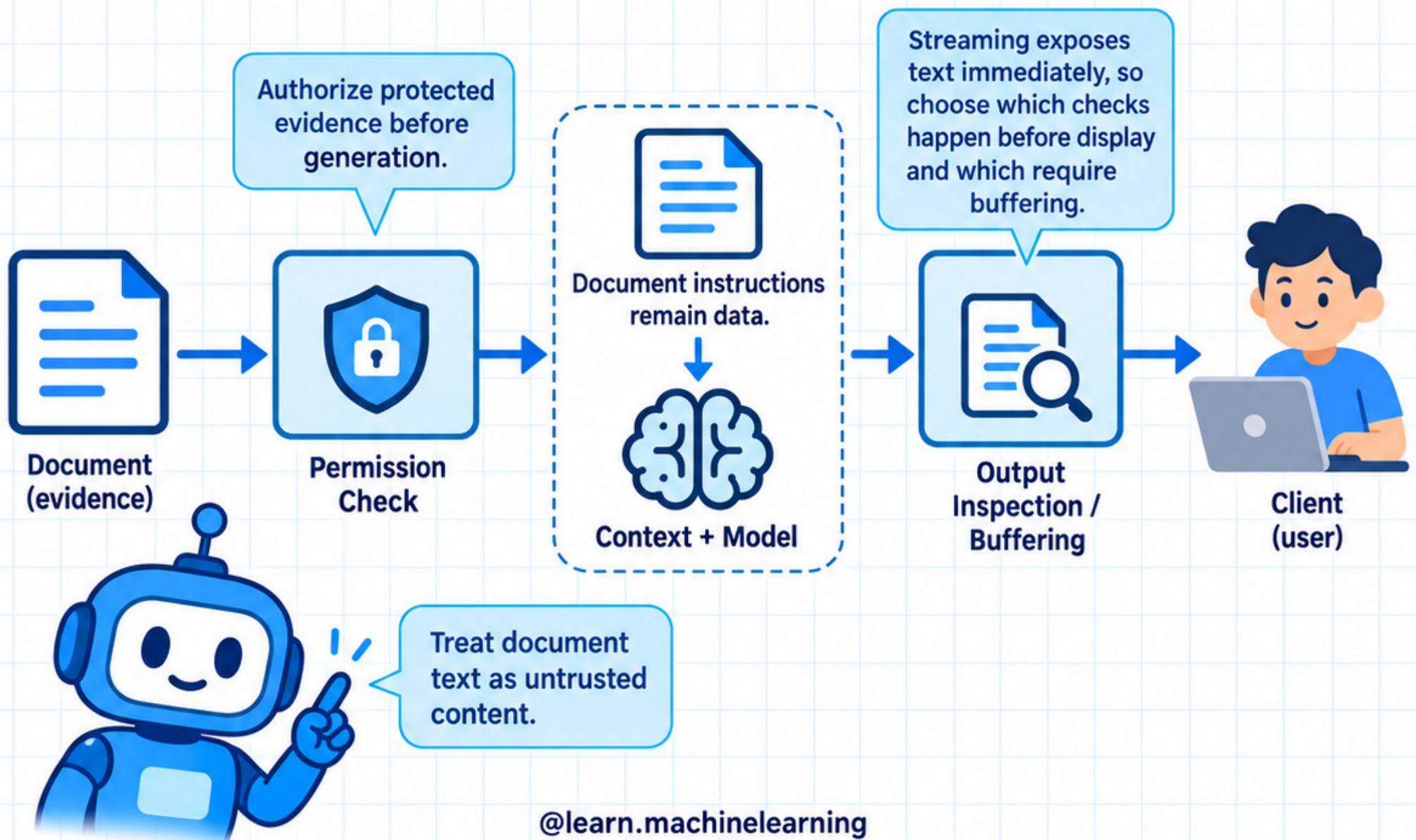
Authorize protected evidence before generation.



Treat document text as untrusted content.



Streaming exposes text immediately, so choose which checks happen before display and which require buffering.



WHERE THE TOOLS FIT

FastAPI



FastAPI StreamingResponse can stream an iterator or async generator.

```
return StreamingResponse(
    event_generator(),
    media_type="text/event-stream"
)
```

Provider API



Provider API: produce documented events.

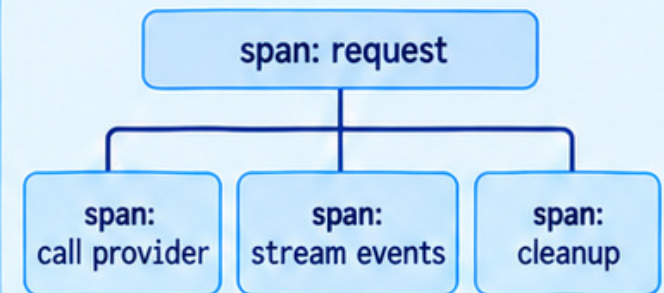
Illustrative; provider formats vary

Content	...
Metadata	...
Terminal status	...

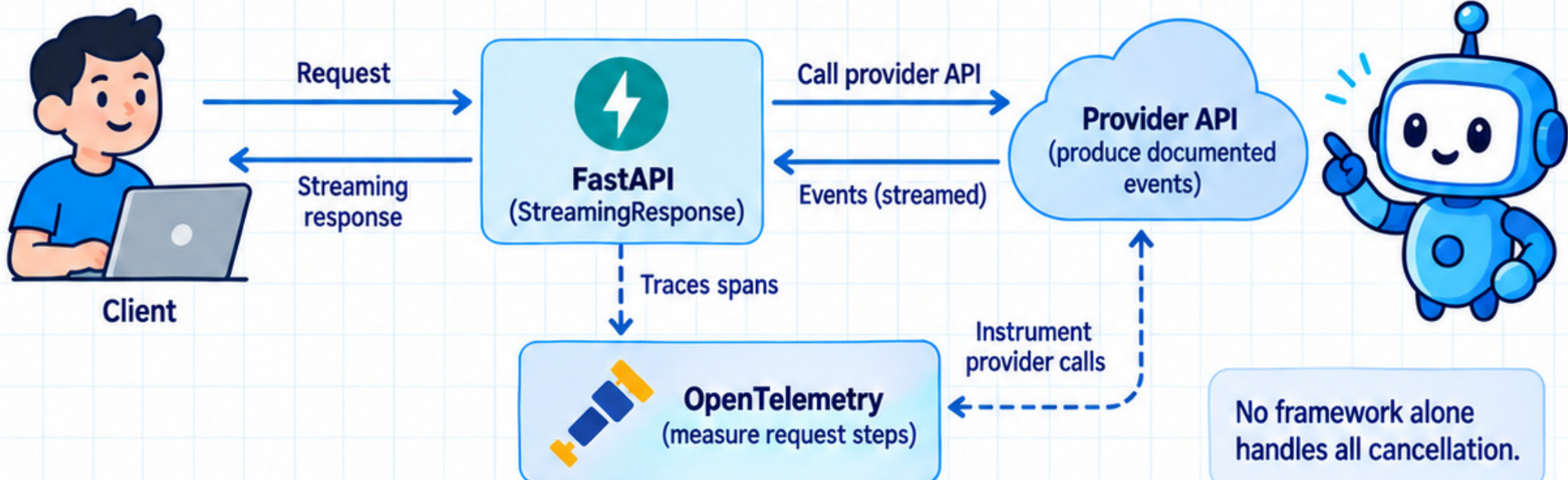
OpenTelemetry



OpenTelemetry spans: measure request steps.



Your code owns event states, cleanup, and failure behavior.

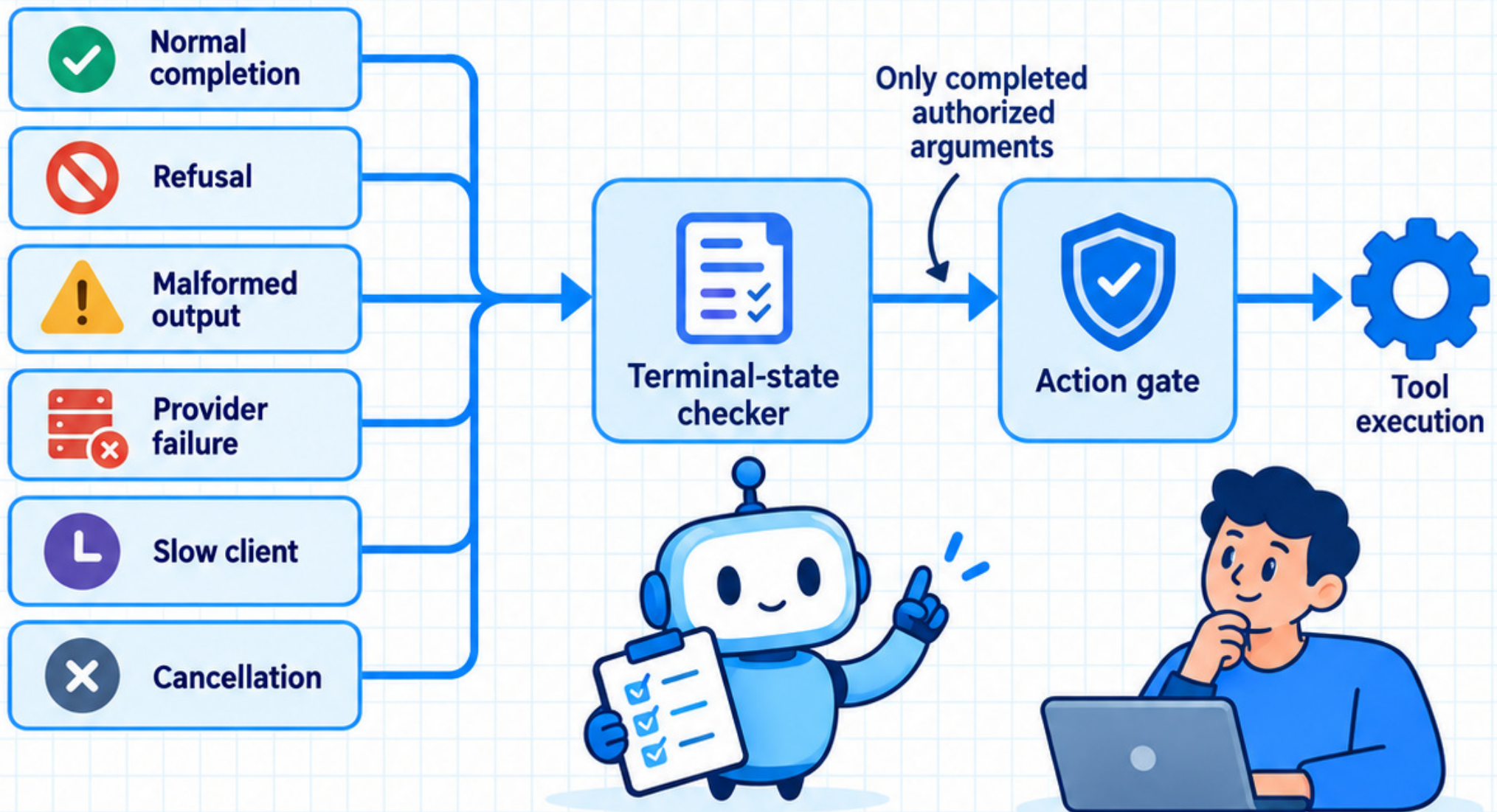


TEST BEYOND THE HAPPY PATH

Test normal completion, refusal, malformed output, provider failure, slow client, and cancellation.

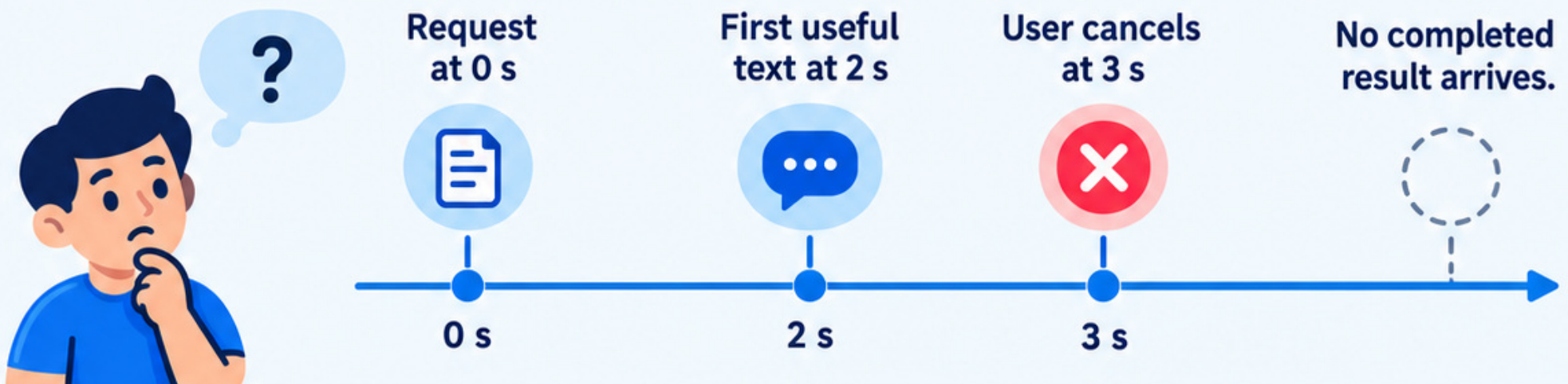
Measure first useful text and usable completion separately.

Check that partial tool arguments never execute.



TRY A CANCELLED STREAM

Toy run: request at 0 s; first useful text at 2 s; user cancels at 3 s.
No completed result arrives.
What latency values and terminal outcome should you record?

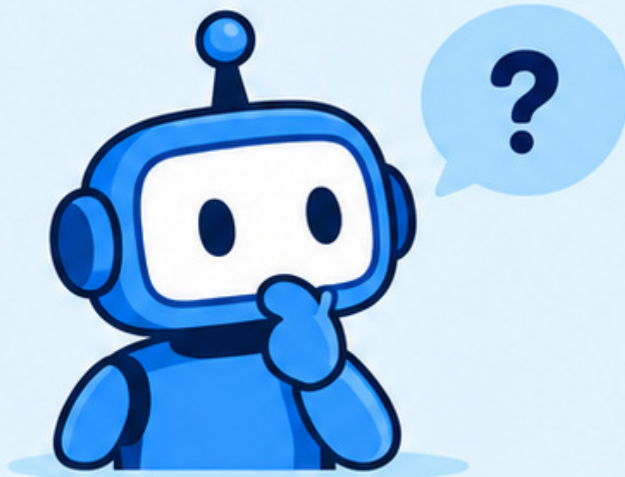


METRIC / OUTCOME	VALUE TO RECORD
Time to first text (s)	—
Time to completion (s)	—
Terminal outcome	—

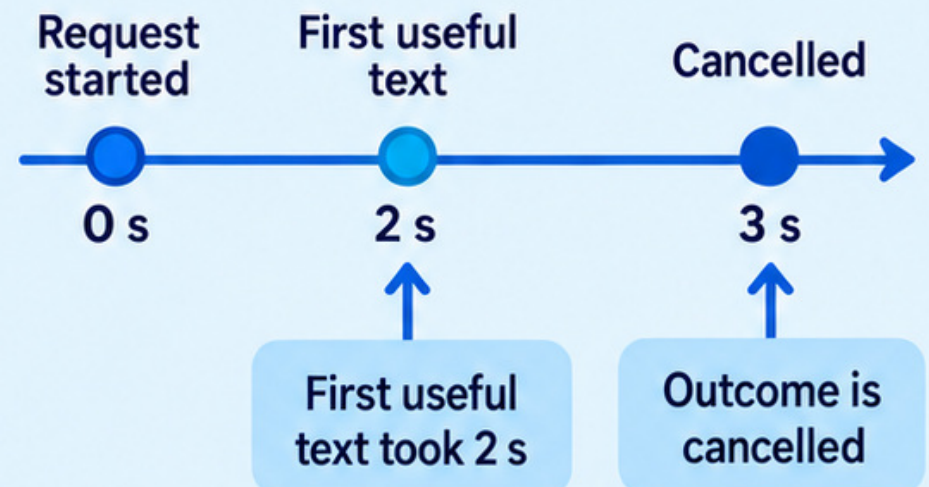


DESIGN THE ENDING TOO

Exercise: first useful text took 2 s; outcome is cancelled. Usable-completion latency is not available.



Record cancellation time separately and include the run in outcome rates.



Next: ingest documents with provenance.



Completed exercise record:

First useful text: 2 s
Usable completion: N/A
Cancellation time: 3 s
Outcome: cancelled

toy