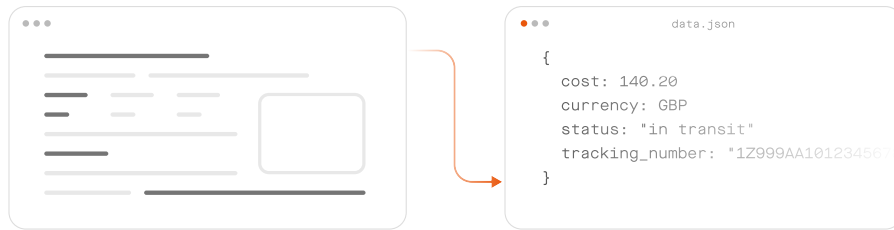# What Are LLM's Useful For?

Forget the hype. Here are the boring, enterprise use cases that are actually raking in the cash.

Structuring Data

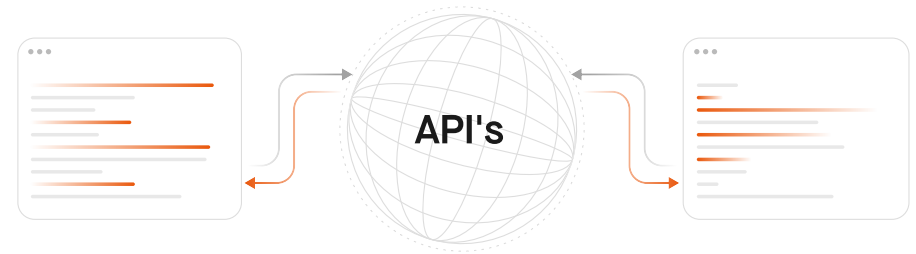Take raw text, images, files, and ingest it into databases.

```
data.json
{
  cost: 140.20
  currency: GBP
  status: "in transit"
  tracking_number: "1Z999AA10123456"
}
```
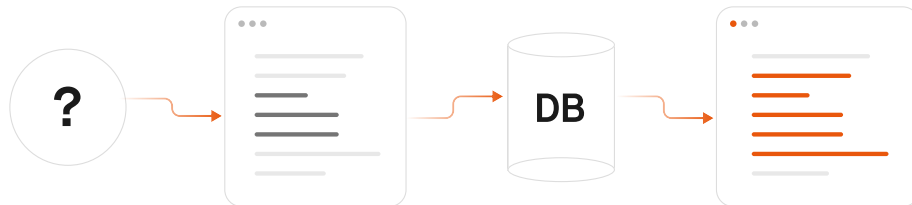
Agents

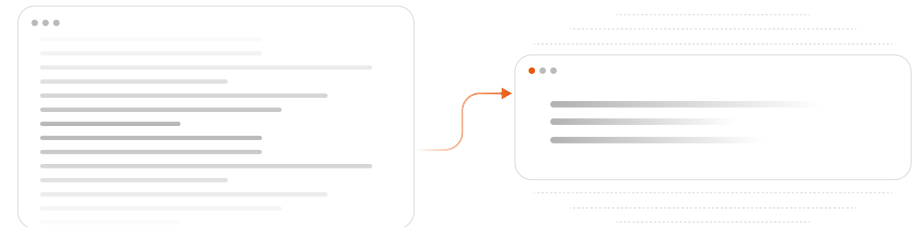Allow an LLM to do things in the world via external API's.

API's

Question Answering

Attach LLM's to external data sources to let them answer questions for users.
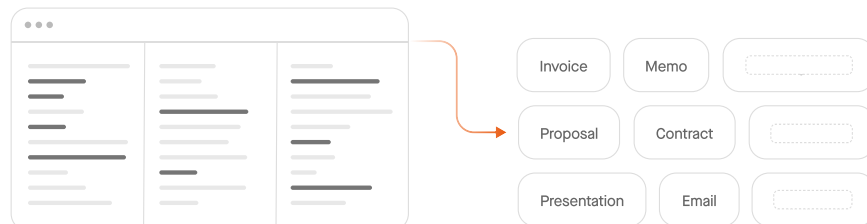
? → **DB**

Summarization

Take longer text and turn it into shorter text. That's it.

Classification & Labelling

Apply meaningful tags to files to make them more searchable. Cheaply categorise things.

Invoice Memo
Proposal Contract
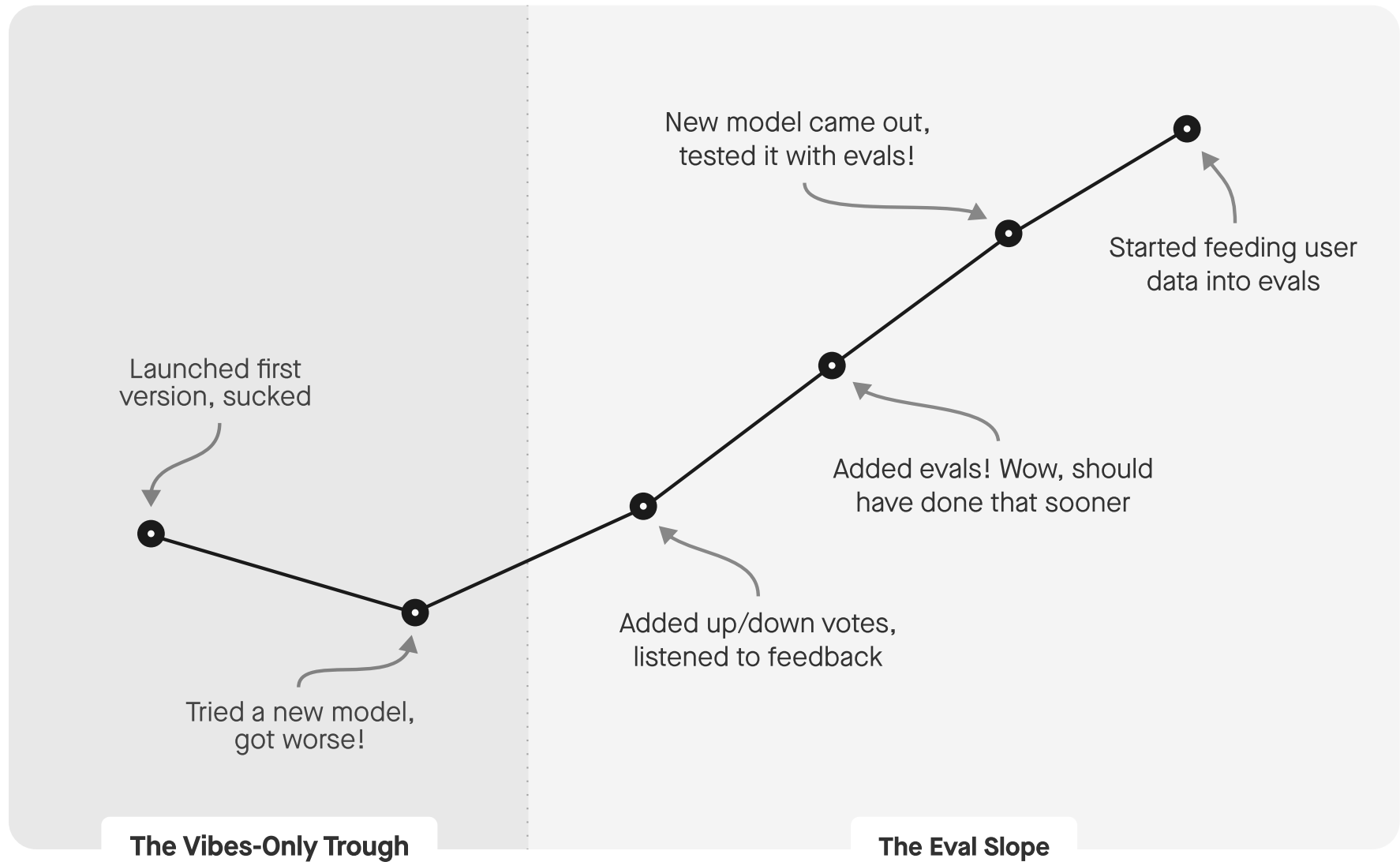Presentation Email

Translation

LLM's are multi-lingual by design, so translation is simple.

How Good AI Apps Get Built

The first version of every AI app sucks. Only user feedback and great evals can improve it.

New model came out,
tested it with evals!

Launched first
version, sucked

Started feeding user
data into evals

Added evals! Wow, should
have done that sooner

Added up/down votes,
listened to feedback

Tried a new model,
got worse!

The Vibes-Only Trough

The Eval Slope

AI Engineering Is Experimental

LLM's are probabilistic, so AI Engineers need to experiment often with new techniques. Those experiments mostly won't work.
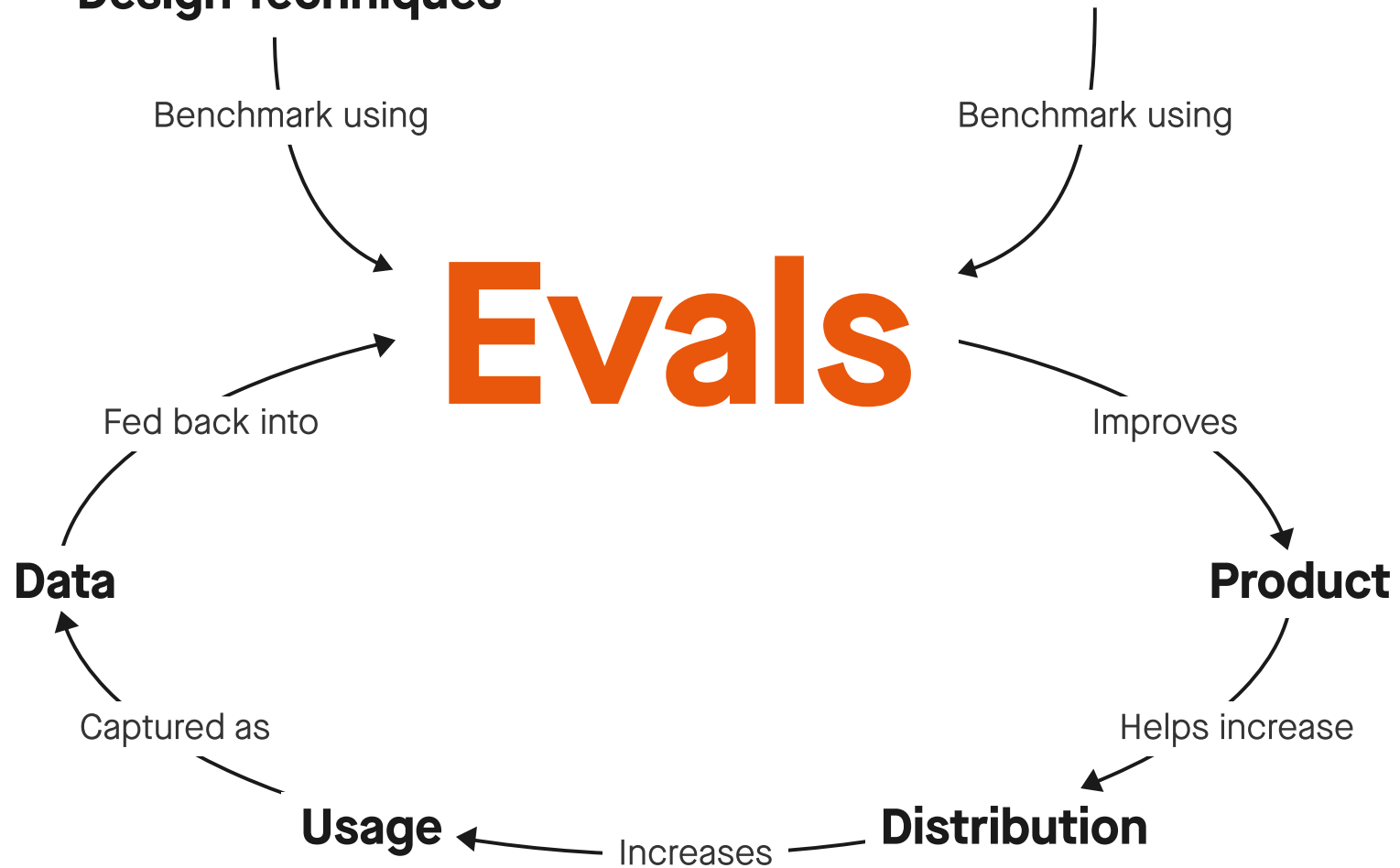
Quality

Time
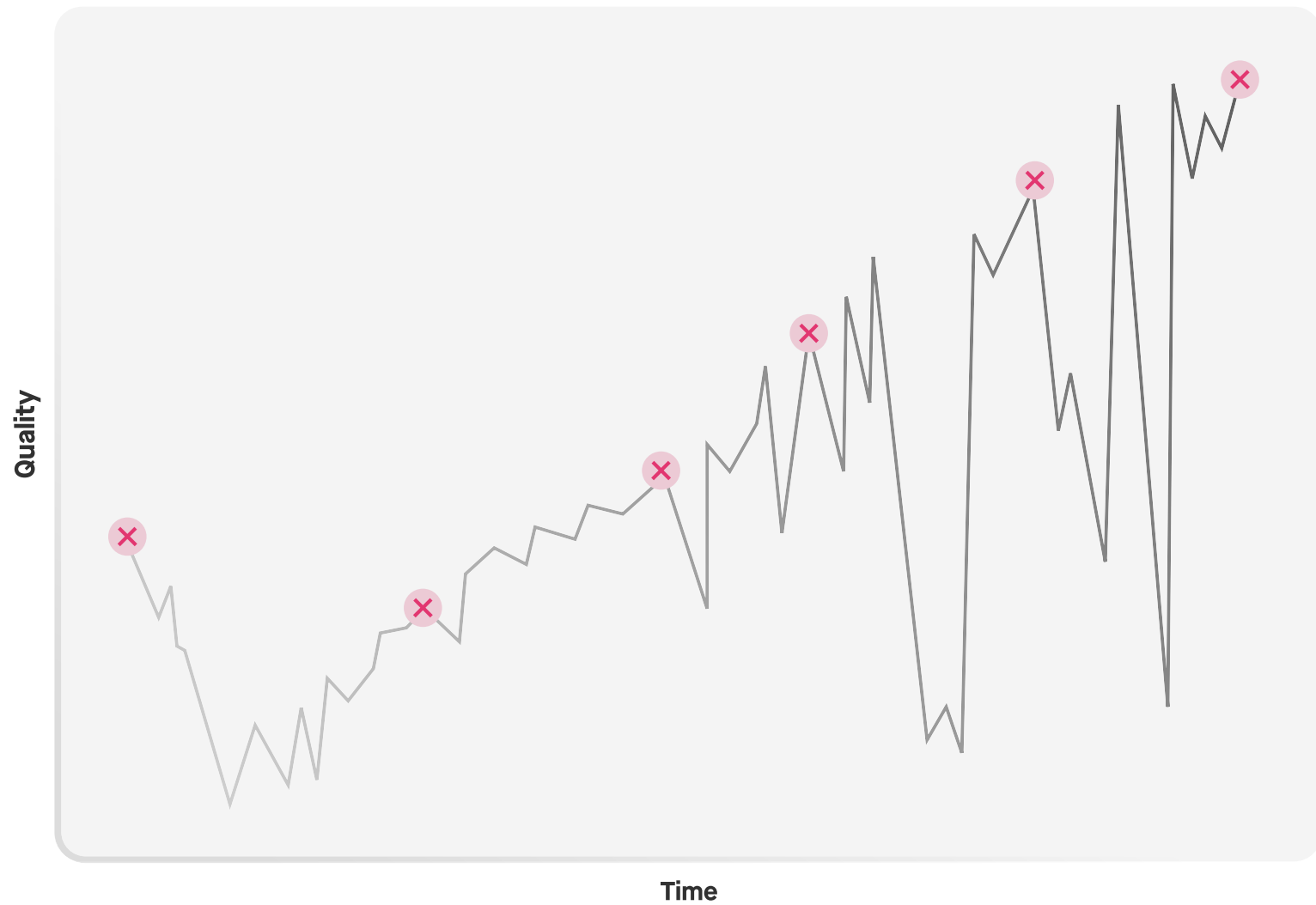
It's The Evals, Stupid

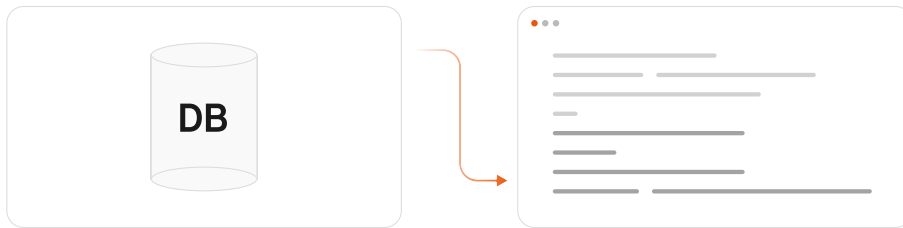Evals are the AI Engineer's unit tests. They help you adapt to new models and fix bad behavior. Without good evals, you're flying blind.

It's All Prompt Engineering

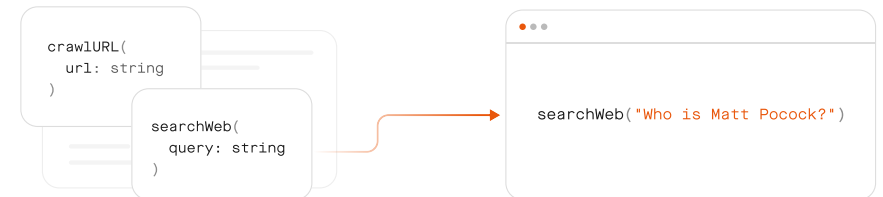All these buzzwords can be confusing. But they're really just improved prompting techniques.

RAG
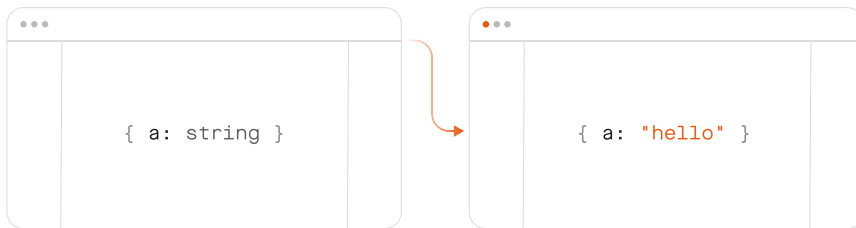
Fetch information from somewhere, put it in the prompt.

Tool Calling

Prompt the LLM with a menu of "actions" it can take in the world.

```
crawlURL(
    url: string
)
searchWeb(
    query: string
)
```

```
searchWeb("Who is Matt Pocock?")
```

Structured Outputs

Prompt the LLM to return a certain shape.

```
{ a: string }
```

```
{ a: "hello" }
```

Workflows

Use multiple prompts split over different LLM calls. Stopping point hard-coded.
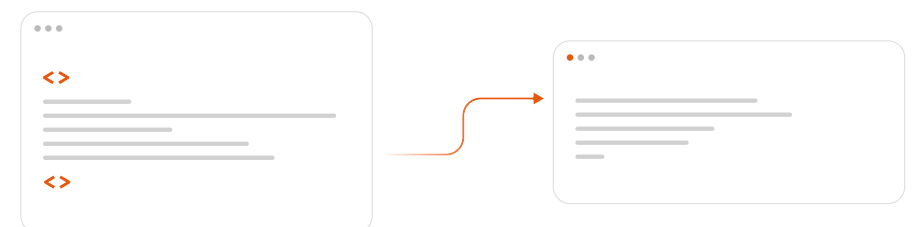

Agentic Loop

LLM interacts with the world via tools, results go back into the prompt. LLM decides when to stop.

Reasoning

Force the LLM to think through its actions via <thinking> tags.

The Staircase To Optimization Hell

Building a new feature? Start at the top of the staircase and walk down. Each step gets more complex and expensive.

Zero-Shot

Few-Shot

Chain of Thought

Temperature

Workflows

Evaluators

Agentic Loops

LLM Routers

Fine-Tuning

Sampling

EXPENSIVE