



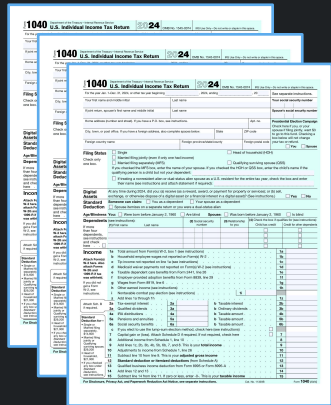
What is required for AI in Government?



A notional IRS example

A complex claim takes 1-2 hours today

Document Package Arrives



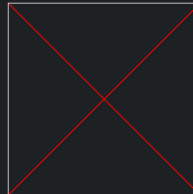
Read & Structure

Read through scanned forms, handwritten fields, & PDFs



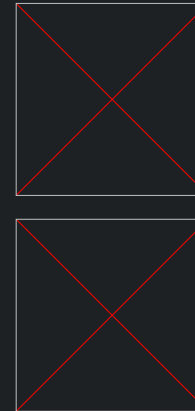
Match to Rules

Check against hundreds of pages of rules



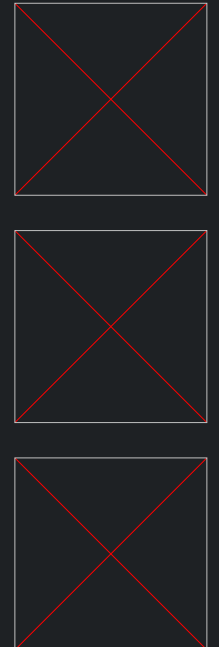
Reconcile with History

Research account information, filings, and past correspondence

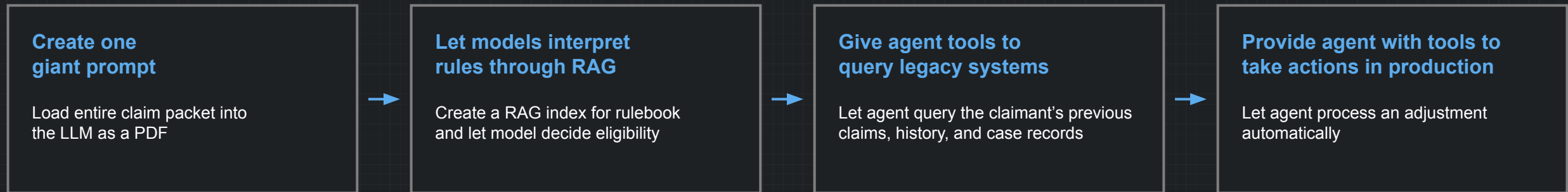


Final Action

Refer to department, send new correspondence, make tax adjustment, etc



The naive build



Execution walkthrough

01—

Run extraction on the whole PDF packet

02—

Extract data and evaluate taxpayer claim

03—

Identify relevant rulebook sections with RAG + semantic search

04—

Call tools to look up account history

05—

Find related form filings and previous correspondence

06—

Translate between written fields and API codes + keywords

07—

Arrive at final decision and take action in production

The naive build problems

1. Agent hits an accuracy ceiling

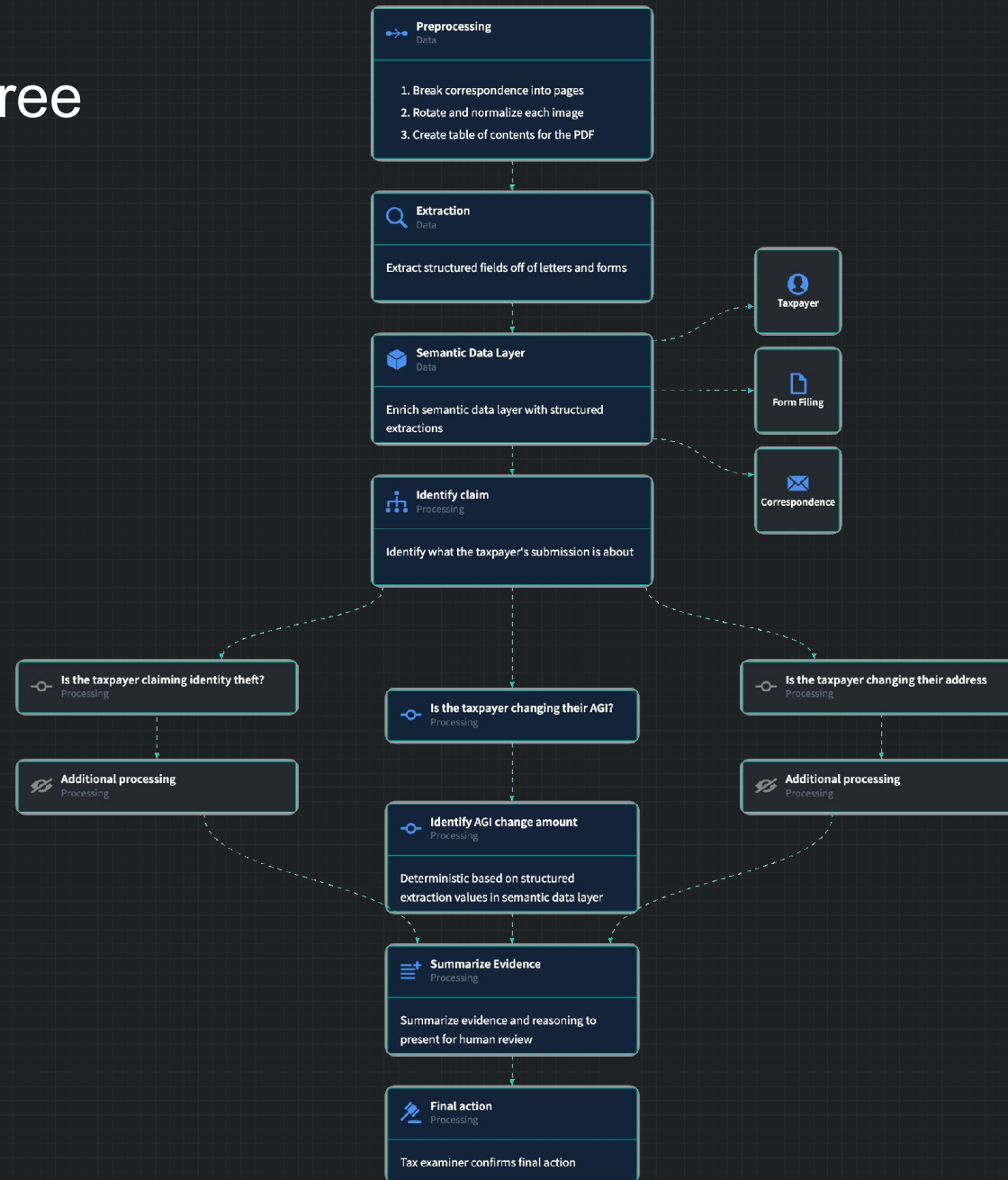
2. Rules are not interpreted consistently

3. Agent struggles with legacy system structure

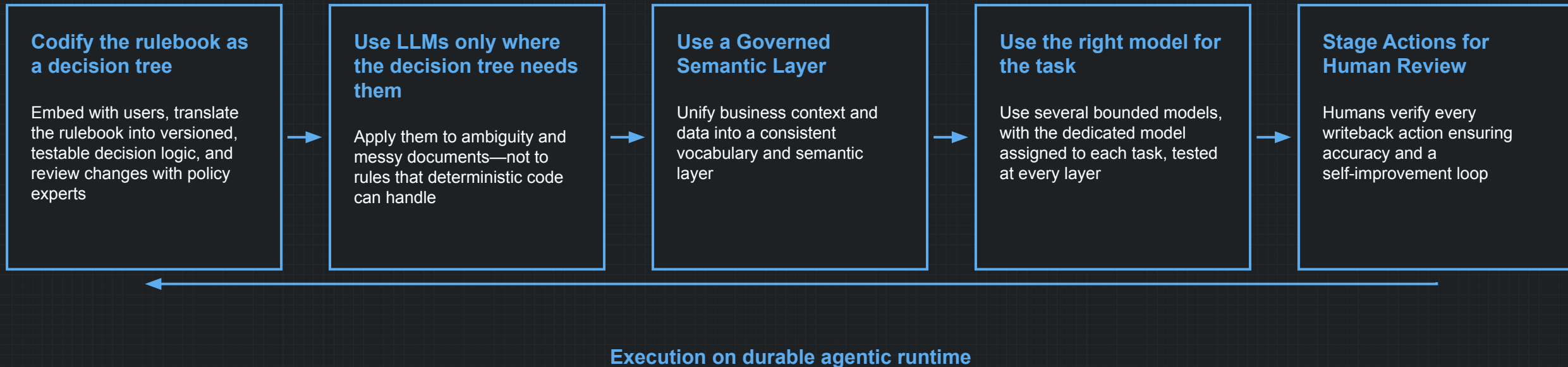
4. Full automation erodes trust

**Architecture
Problem**

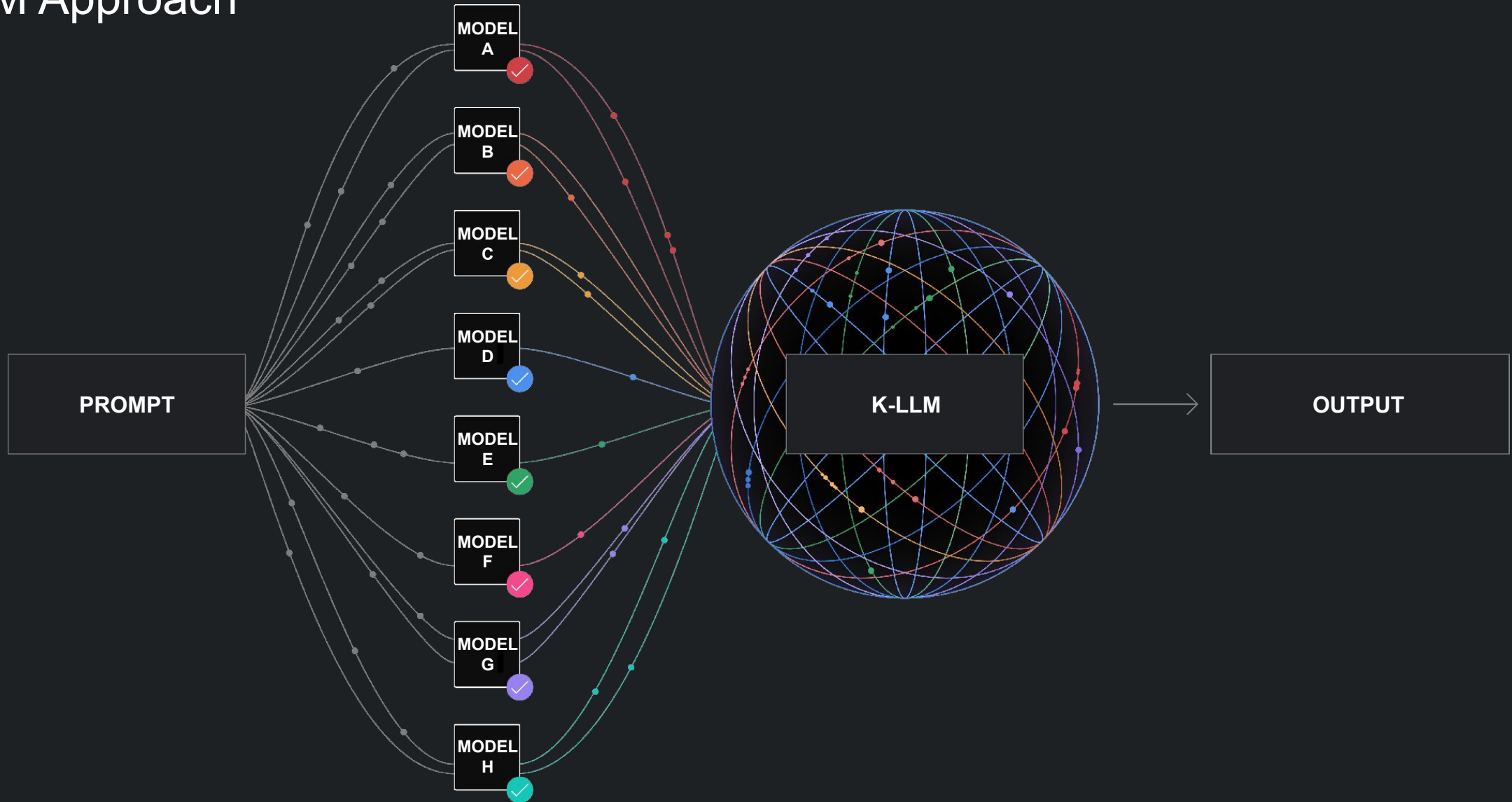
Building the decision tree



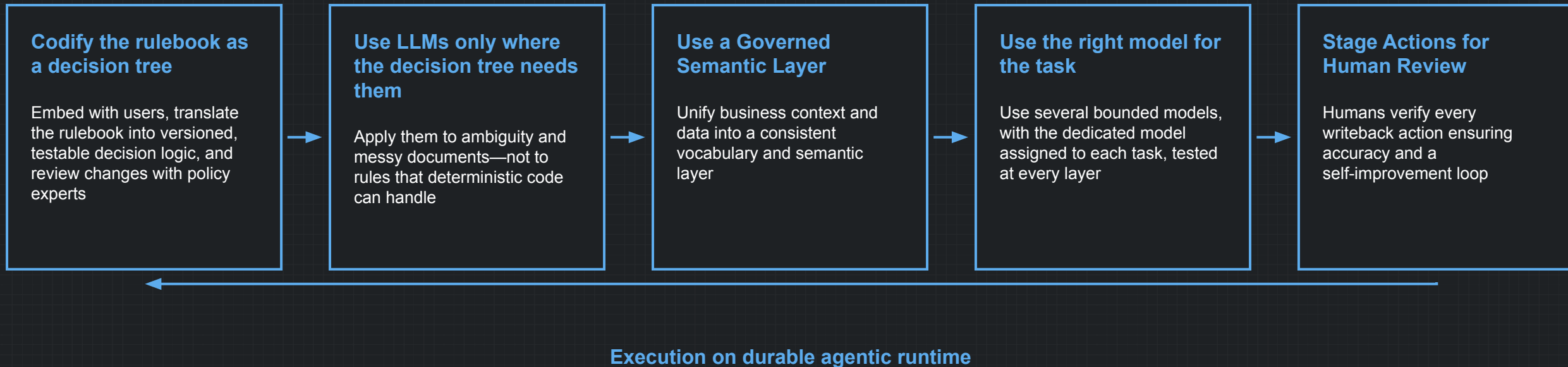
Building it the right way



k-LLM Approach



Building it the right way



Key advantages

Accuracy

Optimizes cost and performance.
Avoids model lock-in.

Determinism

Creates predictability and trust. Enhances
reliability, control, and accuracy.

Traceability

Log everything. Offer reversibility. Feed
into future decisions.

Codify the rulebook as a decision tree

Embed with users, translate
the rulebook into versioned,
testable decision logic, and
review changes with policy
experts



Use LLMs only where the decision tree needs them

Apply them to ambiguity and
messy documents—not to
rules that deterministic code
can handle



Use a Governed Semantic Layer

Unify business context and
data into a consistent
vocabulary and semantic
layer



Use the K-LLM Approach

Use several bounded models,
with the right model assigned
to each task.



Stage Actions for Human Review

Humans verify every
writeback action ensuring
accuracy and a
self-improvement loop

NAIVE BUILD		OUR BUILD
Paraphrased in the prompt	RULES	Codified, versioned, and deterministic
One model does everything	MODELS	K-LLM, with the best bounded model for each task
Different answers	SAME INPUT TWICE	The same answer, with lineage
Improvised tool calls	LEGACY SYSTEMS	Governed connectors & a clearly defined data model
Agent writes to directly production	WRITEBACK	Human reviews and clicks to commit
Vibes	QUALITY	Evaluations against golden datasets
Retries and prayer	FAILURE HANDLING	Checkpoint, retry, and resume
Replaced, allegedly	THE HUMAN	Accelerated, from 2 hrs to 10 mins

