

GROWTH ELEVATED • STAGE III

WHAT IT ACTUALLY TAKES

Building the platform of the future — lessons and examples from the field.

Build the **foundation** first

We used to raise a skyscraper one phase at a time — months of work for a single feature. Now we do it in **days**.

- That speed is exactly why the foundation matters **more**, not less.
- The taller and faster you build, the more the foundation decides whether it stands.
- The pitfall is jumping straight into build-mode — do the legwork first: the real job, and an architecture that holds.

The taller you build, the less you can afford a **cracked foundation.**

You have the **right to win**

The advantage is real — but it doesn't wait for you.

The incumbent's edge

You already own the domain data, the embedded workflows, and years of hard-won customer trust — exactly the raw material AI needs. A newcomer has to earn all of it from zero.

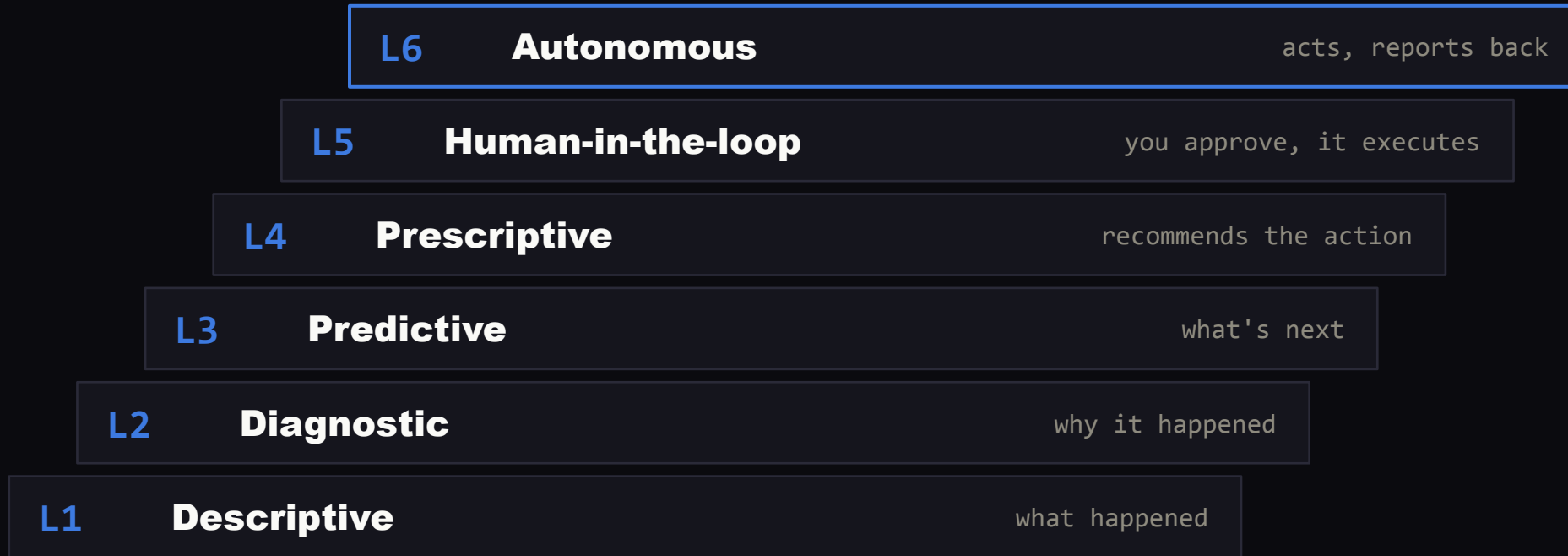
Take it seriously, not fearfully

People really are building against your platforms — the threat is real and worth respecting. But real isn't the same as fatal. The domain depth, integrations, and trust you hold are exactly what's hardest to replicate, so respond with urgency, not fear.

But the clock is running

The edge is real, but it isn't permanent. Move at the pace these tools now make possible, or you eventually get passed. It's yours to lose.

A ladder, not an elevator



WHY YOU CAN'T SKIP

You may not have the data yet — we capture **what** happened, rarely **why**. Climbing is how you find the gaps. Keep the end in mind.

Build the right thing, not **everything**

Knowing what to build was always hard — but the tables have turned.

- **Then:** capacity was the constraint — you prioritized to fit what you could ship in a quarter.
- **Now:** you can build nearly everything — so the discipline is choosing the **right** things, not all of them.

THE TRAP

One company leaned all-in on AI and cleared its entire feature backlog — then watched NPS fall from the high 30s to below zero, because customers couldn't absorb that much change at once. Building everything isn't the same as building the right thing.

Get into the customer's **work**

The real problem lives in the work itself — and you only see it **in person**.

- Over Zoom, they describe today's process — so you just **shave clicks**.
- In the room, you see the **real problem** — the one they can't name.
- Domain expertise **on the payroll** — answers in minutes, not a week's wait.

THE PAYOFF

Get up close and personal, and you finally see what's worth automating — and which steps should just go away entirely.

Who's your user in **two years**?

The mix of humans and agents in your user base decides **how radical** the platform change has to be.

Human-heavy

Agents layer on top of a mostly-human platform. The work looks similar — the change is incremental.

Agent-heavy

One human plus a swarm of agents. A completely different surface — you resurface the platform.

THE IMPLICATION

Where you land sets how radical to go — and where agents are the users, **stop designing for humans** and design for what the agent actually is.

What the future platform **becomes**

You've decided what to build — now how it's delivered. Start with the interface: **layered, composable UX**, not just today's forms.

- **Traditional** — forms & screens, still right for complex, high-stakes work.
- **Agentic** — headless; functionality exposed for agents to call.
- **Dynamic** — views generated on demand, for the moment.
- **Bidirectional agent** — chat and UI interwoven, aware of each other.
- **MCP apps** — your UX running inside Claude or ChatGPT.
- **Customizable** — users reshape the product to their own needs.

What the future platform **becomes**

Beyond the interface — **five more** capabilities an AI-first platform needs.

Proactive labor

Does the work and tells you what to do next — a teammate, not a tool.

Event-based

Reacts the moment something changes.

MCP / API

Exposed so agents can actually act.

Orchestrator-agnostic

Assume it lives outside your walls.

Data as a product

Not just a dataset you claim to own.

You can't build on a **broken foundation**

We used to bolt around the skeletons in our stacks. With agents doing the work, these are the pieces that have to be **solid**.

Clean data model

An agent can map it.

Logic ↔ UX split

No business logic inside a UI.

Single entry point

One door to the logic.

Test coverage

80%+ baseline.

E2E validation

Correct, not just bug-free.

Code analysis

Complexity & duplication gates.

Product telemetry

See how it's really used.

Evals & observability

Agents stay honest.

Design system

Machine-enforceable.

Feature flagging

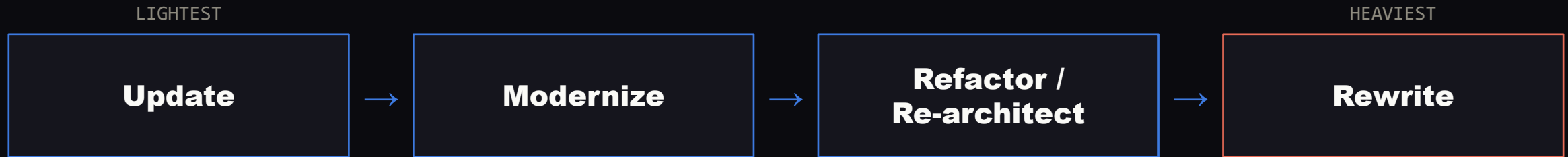
Controlled rollout.

Bursty-load handling

Absorb agent traffic.

The decision at **every layer**

Whether you rewrite is a **data-driven decision** — based on the capabilities you need to support, assessed at every layer.



Non-negotiables for release

Releasing an agent isn't shipping a normal, deterministic feature — non-deterministic behavior needs its own gates.

Evals & observability

Stand up the harness before you launch — no eval layer isn't a launch, it's scheduling an incident.

A dedicated team to watch it

Someone has to own reviewing the agents on a cadence — it outgrows the PM fast.

GA determined by criteria, not a calendar

Go to market on your beta success criteria — a real accuracy bar — not the roadmap date.

Pricing is the litmus test

Ship it free; if they won't pay, it isn't ready — they should be clamoring.

Big takeaways, next steps

CEO

- Treat this as a strategic bet — a **company transformation**, not a roadmap item.
- Fund and protect dedicated capacity — a few big bets each quarter, separate from the day-to-day roadmap.

CPO

- Get in the room with customers — build the **right** thing, not everything.
- Gate GA on **criteria, not a calendar**, with a dedicated owner on evals.
- Define your fully-agentic end state — the job done while they sleep.

CTO

- Make the update → rewrite call **per layer**, data-driven — then fix the foundation.
- Stand up eval & observability infrastructure **before** any agent ships.

GROWTH ELEVATED • WHAT IT TAKES

NOW GO BUILD IT

This isn't a feature you ship — it's the platform you become. Start with the customer; build on a foundation that lasts.

QUESTIONS?