

The Progress of AI.

How one architect turned a folder of Notion agents into an AI operating system for an entire go-to-market org in six months, with security and governance designed into the architecture from day one. And removed himself as the bottleneck along the way.

Quincy Landingham

AI SYSTEMS ARCHITECT · DEFENSESTORM

October 2025 → May 2026

THE THESIS

Democratize *access.* Democratize *information.*

Every workflow in this case study traces back to one question. Who needs the answer, and what's stopping them from getting it themselves?

BY THE NUMBERS · THE CLAUDE SYSTEM · APRIL → MAY 2026

480

HOURS SAVED PER WEEK · Q2
2026 OKR HIT

100+

SKILLS SHIPPED TO
PRODUCTION

2.5

WEEKS FROM PIVOT TO V1
LIVE

3

GTM TEAMS ONBOARDED IN
WEEK ONE · SALES, CS,
ONBOARDING

6

MONTHS START TO FINISH

1

ARCHITECT

CHAPTER ONE

First Proof.

OCTOBER 2025

Black Panther OS goes live as my personal agent orchestrator. The first three applications weren't experiments. They were business decisions.

WIN 1 Winning-deal analysis on a strategic product category. New-hire training rebuilt on the actual patterns of closed-won deals.

WIN 2 Sales knowledge base build accelerated through agent orchestration. Days of content work compressed.

WIN 3 The Fraud Squad analysis. Deep product workup on an underperforming product line, run through Black Panther OS. DefenseStorm retired the product based on this work.

THE ALTITUDE SET ON DAY ONE

AI agents prove their worth when business decisions hinge on them. The first product I shipped to myself informed a product retirement at the company.

CHAPTER TWO

The Fleet Ships.

NOVEMBER → DECEMBER 2025

Two months of experimentation produced one durable pattern. Bring the agent to the user, in the tool the user already opens every day.

EXPERIMENT Agentforce builds shipped. Team didn't adopt at scale.

WIN First teammate-facing Notion agent shipped to an Account Executive. A **multi-mode agent** built on the same orchestration architecture as Black Panther OS, running multiple workflows from a single entry point. The multi-mode pattern became the architectural blueprint I later carried into the production skills system.

WIN Marketing content library agent shipped to a senior marketing leader. Content production lift from solo authorship to agent-orchestrated drafting.

TWO PATTERNS I CARRIED FORWARD

A tool that requires a workflow change doesn't get adopted. The agent goes to the user. And the multi-mode architecture I built here became the blueprint for every skill in the production system. The Notion era taught me how to build the Claude era.

CHAPTER THREE

The Cutover.

JANUARY → FEBRUARY 2026

Title change. Mandate: scale this beyond me. Sixty days, three proofs of leverage.

WIN Board-level emergency churn analysis delivered in 72 hours using Push AI. The kind of question that used to take a week, answered before the next executive meeting.

WIN Invited to join the **Georgian Partners AI Governance working group**. A seat at the table on the broader industry conversation.

WIN Notion agents rolled out to the entire Customer Success organization.

WIN Led the AI agent training session at the Go-to-Market Kickoff (GKO). Coached the Value Squad on using their own agents the same way I had used Black Panther OS for the Fraud Squad. Same playbook, new team.

THE LESSON

Adoption is a teaching problem before it's a tooling problem. Teaching is part of the architecture.

CHAPTER FOUR

Hitting the Wall.

MARCH 2026

First autonomous agents shipped. Office hours started. Then the platform hit its ceiling.

WIN First autonomous Notion agents shipped: a Competitive Intel agent and an AE Onboarding agent. **Both still running today.**

WIN Bi-weekly AI Office Hours launched. A standing room for the team to bring real work and walk out with a working agent.

CEILING Push AI couldn't scale to support the Customer Success signals workflow. The bottleneck was no longer me. It was the platform.

THE DECISION FORMING

Every AI tool on the market promised the layer. None owned the context. I needed a system we could write skills against, on top of our own data, designed for our specific roles.

CHAPTER FIVE

The Pivot.

APRIL 2026

Evaluated the alternatives. Decided to build on Claude.

THE BET

A frontier LLM, our data lake, and a skill library I write specifically for our team will beat anything I can buy. Nobody else can write skills for our team. I can.

WIN v1 built in 2.5 weeks while producing the training content and running the rollout in parallel.

WIN April 20: live for GTM.

WIN Week one of the rollout: three full teams onboarded. Sales (6), Customer Success (4), Onboarding (8). 18 people inside the system in 7 days.

WIN Security and governance shipped as v1 architecture. Output review layers, persona-scoped data access, brand QA, and skill-level guardrails. Built into the system from day one. The only way democratized agent creation is safe at scale.

WIN Office hours moved to weekly. Adoption pulled forward by the people closest to the work.

CHAPTER SIX

The Thesis Lands.

MAY 2026

The OKR was a number. The win was the thesis.

OKR Q2 2026 target: 480 hours saved per week. **HIT.**

WIN First hosted GTM dashboards live: **Protect This House** (customer retention command center), **Pipeline Review** (deal-stage intelligence), **Top of Funnel** (early-pipeline visibility).

WIN 100+ skills in production.

WIN Three executive auto-load contexts shipped (CEO, CMO, Head of Sales).

WIN A full Net-New-Sales operating system shipped to the **entire sales team**: Monday deal review prep, Friday forecast narrative, MEDDPICC qualification rigor, pipeline coverage, and rep scorecards. Every AE running it daily.

THE WIN

The bottleneck is gone. The end users closest to the work are producing the strongest output. That's the thesis. That's the system.

What I'd tell the next architect.

01

Start with business decisions.

The first agent I shipped informed a product retirement. That set the altitude for everything that followed.

03

Be the bottleneck on purpose, once.

Run the workflow yourself before you automate it. The bottleneck is the spec.

05

Find your AI leaders first.

Eager adopters validate the workflow. Their wins do the marketing for free.

07

Ship v1 in days.

Real usage tells you what to build. I shipped a production AI system in 2.5 weeks while running training and rollout.

02

Bring the agent to the user.

Workflow change kills adoption. Meet people in the tool they already open every morning.

04

Build for the least technical user.

Zero prompt engineering required. Smart defaults. Lead with the output, never the prompt.

06

Push, not pull.

The best workflows surface the right thing to the right person without anyone asking.

08

Democratize access. Democratize information.

The whole thesis fits on a sticky note. Every workflow gets measured against it.

The first six months were about *access*. The next twelve are about *creation*.

Democratized access opened the door. What walks through it is work that wouldn't have existed without the system. Dashboards built by the users themselves. Autonomous agents shipped by the people closest to the problem. Operating systems written for one role and adopted by an entire team. All of it running on a security and governance architecture designed in from day one.

What I'm solving for next: closing the distance between a question and its answer, with human-in-the-loop, governance, and secure design as the load-bearing architecture, at scale.

That's where this is going.

Quincy Landingham

AI SYSTEMS ARCHITECT · DEFENSESTORM

Building AI operating systems for go-to-market teams. Skills, MCPs, persona contexts, plugins, push agents, hosted dashboards. End-to-end architecture for AI that the least technical user on the team can actually use.