Case Study: End-to-End Multi-Agent GTM Automation System Built for B2B (Arcus + Client Deployments)

GOAL

Build a fully autonomous, end-to-end GTM system that replaces manual marketing operations for early-stage B2B teams. The objective was to produce strategy, market intelligence, and full-funnel deliverables with near-zero human lift, compressing timelines, reducing dependency on agencies, and giving founders enterprise-grade marketing execution on demand.

The system needed to:

- Think like a strategist
- Produce like a full agency
- Operate like a revenue ops function
- Research like an analyst
- Move at machine speed

This became Arcus, a 50+ agent GTM automation platform, deployed across multiple clients (Numerade, prep for Channel Program, Andesite, and founder-led companies).

WORKFLOW / SYSTEM IMPLEMENTED

1. Multi-Agent GTM Architecture (50+ Agents)

I designed a full-stack orchestration layer that mimicked an entire marketing department, including:

- Strategy agents (positioning, ICP, TAM, segmentation, SWOT)
- Research agents (competitor intelligence, category monitoring, demand mapping)
- Production agents (SEO, landing pages, ads, email, product marketing sheets)
- Analysis agents (attribution, narrative intelligence, funnel diagnostics)

- Validation agents (editorial QC, compliance, channel policy rules, brand checks)

Every deliverable in the GTM stack could be generated or updated through scenario-driven automation.

2. Forensic GTM Engine (Audit System Used Across Clients)

I built a reusable audit engine capable of analyzing:

- ICP patterns
- Channel performance
- Messaging effectiveness
- Competitive positioning
- Funnel gaps
- Opportunity clusters

This engine was used to deliver audit-grade insights for:

- Numerade (edtech)
- Channel Program (cybersecurity/IT channel)
- Andesite (cybersecurity)
- Multiple founder-led projects

It replaced weeks of analyst work with structured, explainable intelligence.

3. Data Ingestion & Intelligence Pipelines

To support autonomous GTM, I built ingestion pipelines using:

- n8n (workflow + automation backbone)
- Claude/OpenAI MCP agents (LLM-executable tools)
- Firecrawl (web crawling & structured extraction)
- Perplexity MCP (autonomous research queries)

These pipelines unified:

- CRM data
- ICP lists
- Competitor site structures
- Keyword graphs
- Organic + paid channel signals

- Social sentiment
- Ad library scans

All data fed into an internal marketing "memory" layer the agents could reason over.

4. Narrative Intelligence & Strategy Loops

The system generated strategy automatically through:

- Narrative intelligence threads (explaining what's happening and why)
- Opportunity mapping (identifying leverage points)
- Automated account scoring (prioritizing targets)
- Insight surfacing (weekly founder updates with recommended actions)

It acted like a chief strategist identifying moves instead of waiting for prompts.

5. End-to-End, Cross-Channel Production Automation

Given a single "scenario pack" (brand, audience, product, constraints), the system could produce:

- SEO topic maps & 8–20 page clusters
- Landing pages (full structure + copy)
- Email sequences
- Paid social/podcast/YouTube ad creatives
- Messaging frameworks
- Persona sheets
- Battlecards
- Product marketing one-pagers
- Social calendar with native-post-ready content
- Demand gen campaigns with audience targeting

Every asset was generated with constraints: brand, tone, compliance, platform rules.

6. Hard Validation & QA Layer

Before anything was "approved," validation agents ran checks for:

- Accuracy and internal consistency
- Platform policies (Meta/TikTok/Google)

- B2B compliance gates (claims, security language, proof points)
- Brand voice + structural standards
- Funnel coherence

This reduced manual editing dramatically — founders could trust outputs.

7. Autonomous Market & Competitor Intelligence

I built ongoing intelligence agents that:

- Monitored competitors
- Tracked category narratives
- Identified breakout patterns
- Surfaced weekly "Here's what shifted, here's what to do" briefs

Instead of static reports, it produced continuously updating insights.

MY ROLE VS ENGINEERS / OPS

My Role:

- End-to-end system architect (50+ agents, orchestration logic, workflow architecture)
- GTM strategist + product thinker (defined ICP logic, funnels, messaging systems)
- PRD owner for every subsystem (architecture → agent behavior → test harness)
- Designed all ingestion schemas, validation layers, and intelligence loops
- Built the entire multi-agent specification inside Cursor/Replit using micro-prompting

Engineers / Ops:

- Implemented my specifications in TypeScript, Python, and n8n
- Connected the orchestrator to MCP servers and tool integrations
- Implemented the UI, routing, execution harnesses, and system monitoring

- Performed refinement tasks based on my test results and validation criteria

I effectively acted as both Head of Marketing and Chief Systems Architect.