Gemini 3 *Edition*

HUMANITY'S LAST PROMPT ENGINEERING GUIDE

by Matthew Berman
& Nick Wentz

FORWARDFUTURE

Humanity's Last Prompt Engineering Guide: Built for the Gemini 3 Era

The Last Guide You'll Need… Until the Next Release

Most prompt engineering guides are stuck in Q2 2025. They either overcomplicate the basics of text-based chatting or give you vague tips that fall apart when you try to build real software or complex agents.

This one is different.

We've done the homework for you, gathering the bleeding-edge best practices from Google's Gemini 3 launch, early access testing of Google Antigravity, and the latest research on agentic behaviors. We have broken it all down into something you can actually use today.

Whether you're in leadership, operations, or engineering, this guide will show you how to stop "chatting" with AI and start orchestrating it. You will learn how to get dramatically better results not just from Gemini 3, but from the entire class of next-generation reasoning models.

How to Use This Guide

This guide is designed to help you become an architect of Ambient Intelligence—using Gemini 3 to control your files, browser, and apps—whether you are a complete beginner or a developer moving into the Antigravity IDE.

Here is what to expect:

- **Section 1:** Introduces the shift from "Chatbots" to "Ambient Intelligence" and why the interface is disappearing.
- **Section 2:** The New Anatomy of a Prompt. We deconstruct how Gemini 3 "thinks" using the new configuration panel: Deep Think, Modality, and Generative UI.
- **Section 3:** The Diagnostic. A rapid-fire guide to diagnosing why an Agent failed (and how to fix it using Cross-Modal Debugging).
- **Section 4:** Breaks down the 6 Core Orchestration Techniques, including Vibe Coding and Antigravity Agent Tasking.
- **Section 5:** Gives ready-to-use Role-Based Templates for the Developer, Analyst, Executive, and Creator.
- **Section 6:** Includes the Gemini 3 Scorecard to evaluate your prompts for "Agency" and "Grounding."
- **Section 7:** A glossary of the new language of AI, from "Nano Banana" to "Context Caching."

How to Use It

- **Skim the Techniques:** If you are a power user, jump straight to Section 4 to learn how to use Antigravity and Deep Think.
- **Use the Diagnostic:** Turn to Section 3 the moment an agent hallucinates or fails to execute a task in the browser.
- **Score Your Work:** Use the Scorecard (Section 6) before deploying any prompt into a production workflow or important meeting.

What You Will Learn

- **Beyond Text:** How to prompt with video, images, and entire codebases using Gemini 3's native multimodal sensors.
- **The "Deep Think" Toggle:** When to pay for expensive reasoning (PhD-level logic) and when to switch to "Low-Think" mode for blazing fast extraction.
- **Vibe Coding:** How to stop writing rigid syntax and start describing the intent and aesthetic you want Gemini to build.
- **Agentic Control:** How to use Antigravity to build, run, and verify software autonomously.

SECTION 1: INTRODUCTION

The Era of Ambient Intelligence

The release of Google's Gemini 3 models marks a definitive turning point in the field of AI, but the most profound shift is not just in the models' staggering capabilities—it's in their ubiquity.

While this guide will delve into the technical mastery of multimodal reasoning and agentic coding within the API, the true power of Gemini is realized as an ambient intelligence. It is no longer an isolated chat box; it is the contextual brain running beneath the surface of your digital life.

With Gemini 3, the best prompt is often the one you barely have to write, as the model automatically synthesizes information across Gmail, Docs, Sheets, Search (AI Mode), and novel developer environments like Antigravity.

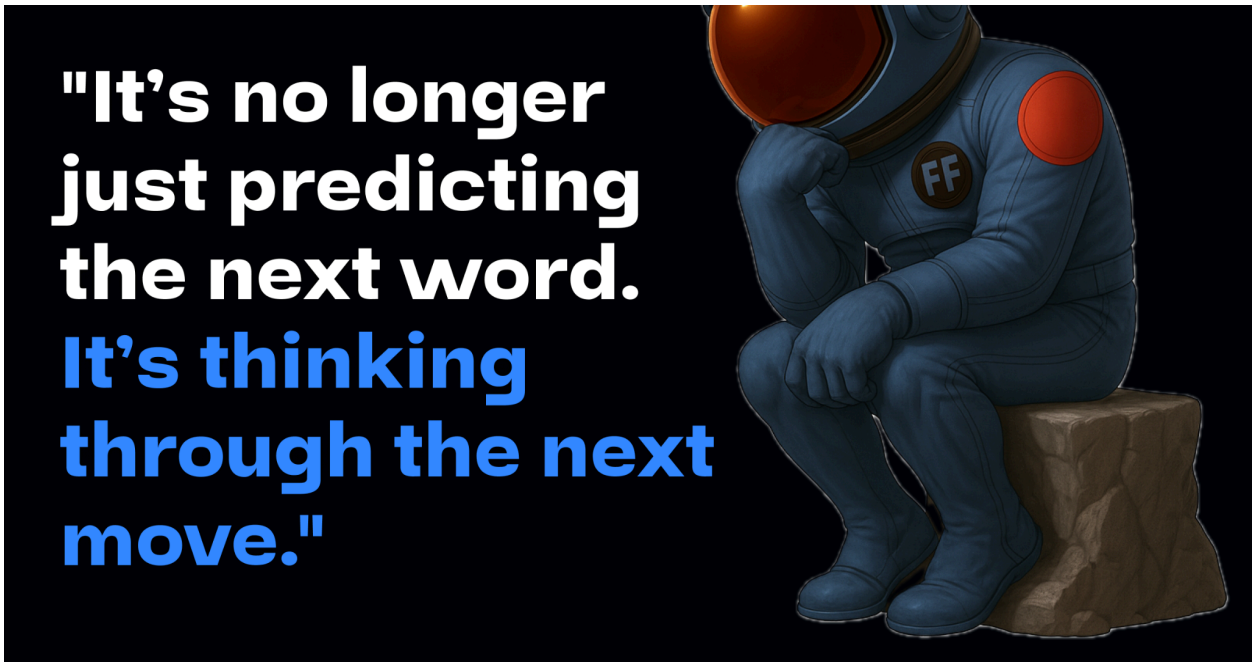
This guide is different. Most guides teach you to write text. This guide teaches you to orchestrate workflows. We've done the homework—testing Gemini Pro, Ultra, and Antigravity agents—to break down how to verify code with images, generate interactive apps on the fly, and manage your calendar with a single sentence.

SECTION 2: HOW GEMINI 3 "THINKS"

Beyond Prediction

In the old world, LLMs were prediction engines—guessing the next word based on text. Gemini 3 is different. It is natively multimodal and agentic.

It doesn't just "read" text; it "sees" charts, "watches" screen recordings, and "feels" the intent behind your code (Vibe Coding). Your prompt is no longer just a query; it is a specification document for an intelligent agent.



The New Configuration Table

Output is affected by more than just words. In the Gemini 3 era, these settings are your control panel:

Setting	Old World Equivalent	Gemini 3 Reality
Thinking Level	Temperature	Controls **Reasoning Depth**. Use `thinking_level="high"` for complex logic; `thinking_level="low"` for fast data scraping.
Modality	Text Input	**Native Multimodal**. You can now prompt with video, audio, and images as primary inputs alongside text.
Grounding	Copy/Paste	**Ambient Context**. Tag files (`@Q3_Report`) or apps (`@Gmail`) directly. The model "lives" in your workspace.
Output	Text/Code Blocks	**Generative UI**. Force the model to build interactive tools (calculators, maps, forms) instead of static text.

SECTION 3: THE NEW ANATOMY OF A PROMPT

From "Context Dumping" to "Ambient Referencing"

In the early days of AI (2023-2024), you had to be a "Prompt Engineer." You had to type out long, detailed paragraphs—known as Context Dumps—to explain who you were, what you wanted, and why it mattered. You had to explicitly answer the 5 Ws: Who, What, Where, When, Why.

Gemini 3 changes the anatomy. Because Gemini is Ambient (connected to your Docs, Calendar, and Photos) and Multimodal (can see and hear), you no longer need to type the context. You just need to point to it.

The Shift: The "Perfect Hike" Example

Let's look at how a request evolves from the Old World to the Gemini 3 World.

🔴 The Old Way (The "Context Dump")

Reliant on you typing everything perfectly.

User: "My girlfriend and I are avid hikers. We live in SF. We've done Mt. Tam and the Presidio. We want something 2 hours away, medium length, unique views, and a breakfast spot at the end. We are free this weekend. Please recommend a hike."

Critique: This is fragile. If you forget to mention you hate steep inclines, the model fails. It relies entirely on your memory and typing speed.

🟢 The Gemini 3 Way (Ambient Referencing)

Reliant on facts and existing data.

User: "Plan a hike for us. Context: Check @Calendar for our shared free slot this weekend. Preferences: Look at my @Google_Photos album 'Favorite Hikes 2024' to see the terrain we like. Constraints: Must be within 2 hours of SF, but exclude any trails listed in my @Completed_Hikes sheet. Goal: Find a trail with a similar 'vibe' to the photos but with a 4.5+ star breakfast spot nearby."

The New 5 Ws

In Gemini 3, you don't answer the questions yourself; you connect the data sources that hold the answers.

Question	Old World Answer	Gemini 3 Answer
WHO	"I am an avid hiker..."	**@Google_Photos** (Visual proof of your skill level and terrain preference).
WHEN	"This weekend..."	**@Calendar** (Exact, real-time availability).
WHAT	"A medium hike..."	**@Completed_Hikes** (Data-driven exclusion of what you've already done).
WHY	"For a unique adventure..."	**"Vibe Match"** (The model analyzes your photos to understand what "unique" means to *you*).
WHERE	"2 hours from SF..."	**AI Mode in Search** (Real-time traffic and travel calculation).

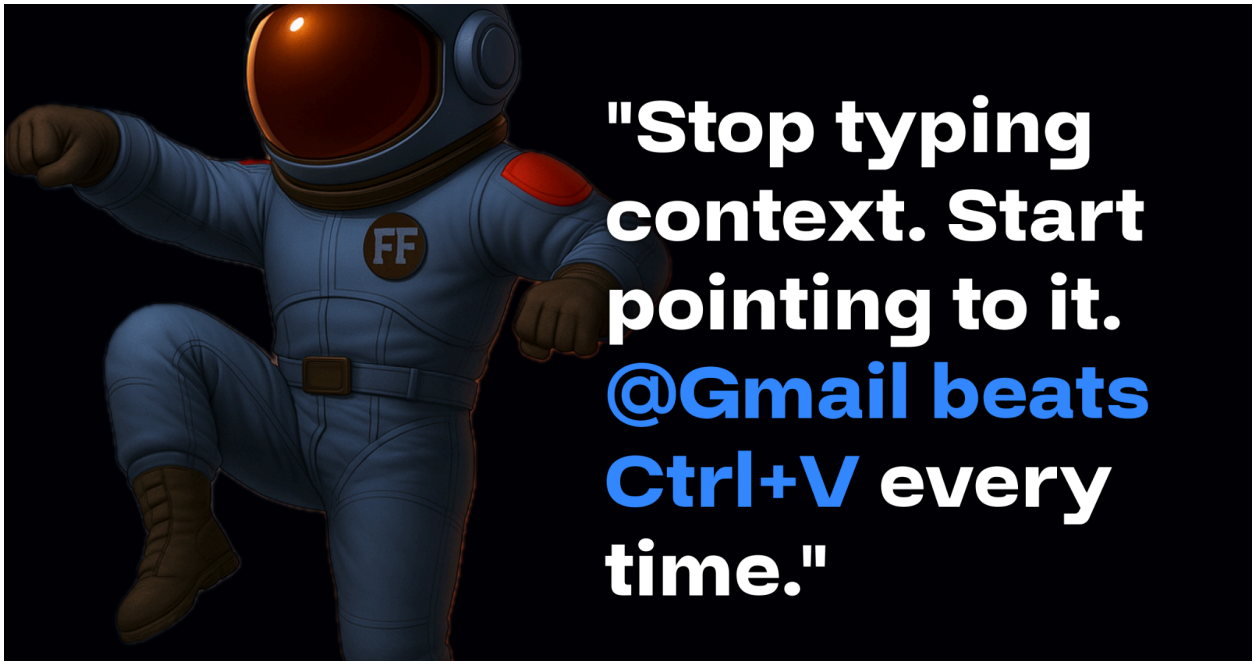
Why This Matters

"Context Dumping" forces the model to guess based on your description. Ambient Referencing forces the model to ground its answer in reality.

When you use Gemini 3, stop asking: "How do I describe this?" Start asking: "Where does this information already live?"

- Is the "Context" in an email thread? Tag @Gmail.
- Is the "Style" in a PDF presentation? Tag @Drive.
- Is the "Problem" on your screen? Use the Video/Screen Sensor.

The perfect Gemini 3 prompt isn't a novel; it's a connection request.



SECTION 4: THE DIAGNOSTIC

Fixing Broken Prompts in the Multimodal Age

Bad output isn't always the model's fault. It's often a failure to use the right sensors (eyes/ears) or tools.

Problem	Weak Prompt (Old World)	Gemini 3 Fix
No Visual Context	"Why is this code throwing an error?"	**Cross-Modal Debugging:** "Analyze this **screenshot** of the error log alongside the code file. Identify the UI element causing the crash."
Over-Explanation	"Think step-by-step and list pros and cons..."	**Parameter Control:** "Compare these options using `thinking_level='high'`. Output the optimal choice immediately."
Static Output	"Calculate the monthly cost of this loan."	**Generative UI:** "Build me an **interactive mortgage calculator** that lets me slide the interest rate to see the monthly impact."
Manual Labor	"Here is the email text. Draft a reply."	**Ambient Agent:** "Check **@Gmail** for the last thread with Client X. Draft a reply based on our agreed timeline."

SECTION 5: THE 6 CORE TECHNIQUES

Mastering the Gemini 3 Ecosystem

Prompting is no longer just about writing clever sentences. It is about Orchestration. These six techniques focus on high-leverage Gemini 3 capabilities.

1. Cross-Modal Debugging

What it is: Using the model's eyes to verify code or logic. When to use it: When a script runs "successfully" but the output looks wrong (e.g., a broken chart or UI glitch).

> ***The Prompt:*** *"Analyze the provided image (screenshot of a chart). Cross-reference it with the adjacent Python code. Identify the bug in the code that caused the Y-axis mislabeling shown in the image, and provide the corrected line."*

⚡ **Why it works:** Standard linters only check syntax; they can't see if a chart looks "wrong." By giving Gemini visual access, you force it to reconcile the **visual ground truth** (the image) with the logical instructions (the code), allowing it to spot semantic errors that are technically "bug-free" but visually broken.

2. Antigravity Agent Tasking

What it is: Delegating a full engineering loop—plan, code, verify—rather than asking for a snippet. When to use it: Building features in the Antigravity IDE where the model has terminal access.

> ***The Prompt:*** *"As the lead agent, fully implement client-side validation for the 'Contact Us' page. After implementing, run the server and generate an Artifact (screenshot) showing the successful validation error state. Report on: 1. Plan, 2. Execution, 3. Verification."*

⚡ **Why it works:** Traditional prompts generate a "best guess" snippet. Antigravity agents create an **autonomous feedback loop**: they write the code, execute it in a real terminal, observe the error, and self-correct before presenting you with the final result. It replaces human trial-and-error with machine recursion.

3. Generative UI (GenUI)

What it is: Forcing the model to render an interactive application instead of text. When to use it: When you need to simulate a decision, compare complex data, or visualize a result.

> ***The Prompt:*** *"I am deciding between two mortgage options (A: 6.5%/2pts, B: 7.1%/1pt). Generate an interactive loan calculator that lets me input loan amounts and see the cost difference side-by-side."*

⚡ **Why it works:** Humans struggle to compare complex tradeoffs (like interest rates) in text format. GenUI shifts the cognitive load from reading to simulating. Instead of explaining the math, the model builds a bespoke tool that lets you "feel" the data by interacting with it directly.

4. Parameter-Aware Extraction

What it is: Using thinking_level="low" to strip away "thought" for pure speed and extraction. When to use it: Scraping data from documents where you don't need analysis, just facts.

> ***The Prompt:*** *"Extract only the CEO name and Revenue from these 20 PDFs. Do not perform any reasoning. (Use thinking_level='low' configuration)."*

⚡ **Why it works:** Complex reasoning models (Deep Think) are "overqualified" for data scraping, wasting time and compute on unnecessary logic. Setting `thinking_level="low"` bypasses the model's internal Chain-of-Thought circuitry, turning it into a hyper-fast, deterministic pattern matcher.

5. The Workspace Agent (Ambient Orchestration)

What it is: Connecting Gemini to your Google Workspace (Docs, Gmail, Calendar) to perform real-world actions. When to use it: Scheduling, email management, and project coordination.

> ***The Prompt:*** *"Review my last 5 emails from 'Client X'. Find the consensus meeting time. Check my @Google Calendar for availability. If free, book the slot and draft a confirmation reply."*

⚡ **Why it works:** It eliminates the "Context Dump." By using **Ambient Referencing** (`@Gmail`, `@Calendar`), the model grounds its response in your live data rather than its training data. It doesn't hallucinate a meeting time because it is literally looking at the empty slot on your calendar grid.

6. Vibe Coding (Role Refined)

What it is: Describing the intent and aesthetic of code rather than technical syntax. When to use it: Rapid prototyping or when you want a specific engineering culture (e.g., "Ship it fast" vs. "Enterprise grade").

> ***The Prompt:*** *"You are a pragmatic startup engineer. Build this landing page. Prioritize speed and visual impact over perfect architecture. Use Tailwind for styling. It should feel 'playful but trustworthy.'"*

⚡ **Why it works:** LLMs excel at **latent space mapping**—translating abstract concepts into concrete specifications. The model understands that "playful" statistically correlates with rounded corners, vibrant gradients, and bounce animations, while "trustworthy" maps to serifs and blues, saving you from writing hundreds of lines of CSS.

SECTION 6: ROLE-BASED TEMPLATES

Ready-to-Use Gemini 3 Workflows

These templates are not just "questions"—they are agentic instructions. They leverage specific modes (Antigravity, Deep Think, Ambient) to perform complex work.

👩‍💻 The Developer

Primary Mode: Antigravity (Agentic IDE) | Secondary Mode: Vibe Coding

1. The Legacy Refactor (Antigravity)

"Ingest this entire GitHub repository. Identify the three most brittle functions based on cyclomatic complexity. Refactor them to use modern async/await patterns. Then, use the terminal to run the existing test suite and generate an Artifact (screenshot) confirming no regressions."

2. The "Vibe Code" UI Build (Vibe Mode)

"You are a pragmatic frontend engineer. Build a 'Waitlist' landing page. Vibe Check: It should feel like a 'cyberpunk terminal'—monospaced fonts, neon green borders, glitch effects on hover. Prioritize visual impact over clean architecture. Deploy it to a local preview server."

3. The Test Architect (Antigravity)

"Scan the @/src/components folder. You will see 12 components without test files. Write a comprehensive Jest test suite for each one, covering positive, negative, and edge cases. Run the tests and auto-fix any failures until we reach 100% coverage."

4. The Documentation Bot (Multimodal)

"Watch this screen recording of me clicking through the new checkout flow. Simultaneously read the checkout.ts code. Generate a Markdown documentation file explaining how the user flow maps to the backend logic, including a sequence diagram."

📊 The Analyst

Primary Mode: Deep Think | Secondary Mode: Generative UI

1. The Strategic Audit (Deep Think)

"Activate Deep Think. Watch the uploaded video of the Competitor's Keynote. Cross-reference their spoken claims with the financial data in their attached Q3 PDF Report. Highlight three specific instances where their marketing narrative contradicts their actual R&D spend."

2. The Scenario Simulator (Generative UI)

"I need to price our new SaaS tier. Generate an interactive pricing calculator that lets me toggle variables (CAC, Churn Rate, Price Per Seat) to visualize the impact on Net Revenue Retention (NRR) over 24 months. Make it a slider-based UI."

3. The Root Cause Detective (Deep Think)

"We saw a 15% drop in conversion yesterday. Review the @Server_Logs and the @Customer_Support_Tickets from the last 24 hours. Deep Think: Correlate the timestamp of the server errors with the sentiment spike in tickets. What exactly broke, and who is most affected?"

4. The Visualizer (Multimodal)

"Take this messy raw CSV of Q4 sales data. Generate a distinct interactive heatmap showing sales density by region. Overlay a line graph showing year-over-year growth. Allow me to hover over regions to see the top-performing sales rep."

🚀 The Executive

Primary Mode: Ambient Intelligence | Secondary Mode: Deep Think

1. The "Who is this?" (Ambient)

"I have a meeting with 'Sarah Jenkins' in 5 minutes. Scan @Gmail and @Drive for every interaction we've ever had. Summarize: 1) How we met, 2) Her last three requests, and 3) The status of our current project. Output as a 1-minute briefing bullet list."

2. The Inbox Triage (Agentic)

"Review my unread @Gmail from the last 24 hours. Archive anything that is a newsletter or cold outreach. Flag the 3 emails that require a strategic decision from me personally. Draft short, affirmative replies to the rest proposing a meeting next week."

3. The Decision Stress-Test (Deep Think)

"I am about to approve the 'Project Titan' budget. Read the attached Proposal PDF. Play Devil's Advocate. Give me the top 3 reasons why this project will fail, based on our company's historical constraints found in @Post_Mortem_2024."

4. The Board Prep (Multimodal)

"Look at my @Calendar for the past quarter to see where I spent my time. Compare it to our @OKRs_2025 document. Generate a pie chart showing 'Time Invested vs. Strategic Priority.' Draft a paragraph explaining any misalignment."

🎨 The Creator

Primary Mode: Nano Banana (Image/Video) | Secondary Mode: Vibe Coding

1. The Asset Orchestrator (Nano Banana)

"I need a hero image for the new 'Eco-Friendly' campaign. Take this product shot of our bottle. Use Nano Banana to: 1) Remove the studio background, 2) Place it on a mossy rock in a rainforest, 3) Add a soft 'morning mist' lighting effect. Output 3 variations."

2. The Content Repurposer (Multimodal)

"Watch my latest YouTube Video (attached). Extract the three most punchy 30-second segments. Transcribe them and draft a LinkedIn post for each, using a 'Thought Leader' tone. Then, generate a thumbnail image for each post using the video's keyframe."

3. The Brand Police (Deep Think)

"Review this drafted blog post against our @Brand_Voice_Guidelines. Identify every sentence that sounds too 'corporate' or 'passive.' Rewrite those sentences to sound punchy, authoritative, and human. Explain why you changed them."

4. The Interactive Teaser (Generative UI)

"We are launching a mystery product. Generate a cryptic interactive web module that reveals one clue every time the user clicks a pixel. It should have a 'noir detective' aesthetic and end with a countdown timer to Friday."

SECTION 7: THE GEMINI 3 SCORECARD

Evaluating Your Prompts

Prompt engineering is iterative. Use this checklist to ensure you are using the full power of the model.

Question	Score (1-5)
1. Ambient Check: Did I tag relevant files/apps (@Gmail, @Docs) instead of pasting text?	
2. Modality Check: Did I include a screenshot, video, or chart to "ground" the model?	
3. Agency Check: Did I ask for an *Artifact* (proof of work) rather than just text?	
4. Reasoning Check: Did I correctly toggle **Deep Think** (On for logic, Off for speed)?	
5. GenUI Check: Did I ask for an interactive UI element if a tool would be better than text?	

Scoring:

- **20-25: Superuser.** You are orchestrating an intelligence.
- **10-19: Proficient.** Good, but you might still be treating it like a chatbot.
- **0-9: Legacy.** You are using Gemini 3 like it's 2023. Update your workflow.

SECTION 8: GLOSSARY

The Language of Gemini 3

- **Ambient Intelligence:** AI that runs in the background of your OS/Workspace, "reading the room" and acting on your files without manual uploading.

- **Antigravity:** Google's agentic development platform where Gemini acts as an autonomous coding agent with full terminal/browser access.

- **Computer Use (Gemini 2.5):** A specialized model capability that allows agents to control a web browser (clicking, typing, scrolling) to verify tasks.

- **Deep Think:** Gemini 3's enhanced reasoning mode, capable of PhD-level logic and solving novel challenges (like ARC-AGI-2).

- **Generative UI (GenUI):** When the model generates a clickable, interactive interface (buttons, sliders, forms) as its response.

- **Nano Banana:** The internal name for the high-performance image editing model available within Antigravity agents.

- **Vibe Coding:** A prompting style that focuses on the "feel" and intent of the software, which Gemini 3 translates into code.

FINAL THOUGHT

The End of the "Prompt"

Prompting is no longer about tricks, syntax, or finding the perfect magic word. It is about orchestration.

With Gemini 3, the rigid text box is dissolving. The model isn't just waiting for you to type; it is "reading the room," analyzing your workflow, and waiting for permission to act. You are no longer a user chatting with a bot—you are a Director managing a team of agents across Antigravity, Search, and your Workspace.

The era of "talking" to AI is over. The era of building with it has begun.

This guide gave you the blueprint. Now go forth and take back your time.

FORWARDFUTURE

Helping everyone benefit from AI with timely news & accessible education.

Website: forwardfuture.ai

YouTube: @matthew_berman

X: @forwardfuture

Thanks for reading!

- Matt, Nick, and the Forward Future Team