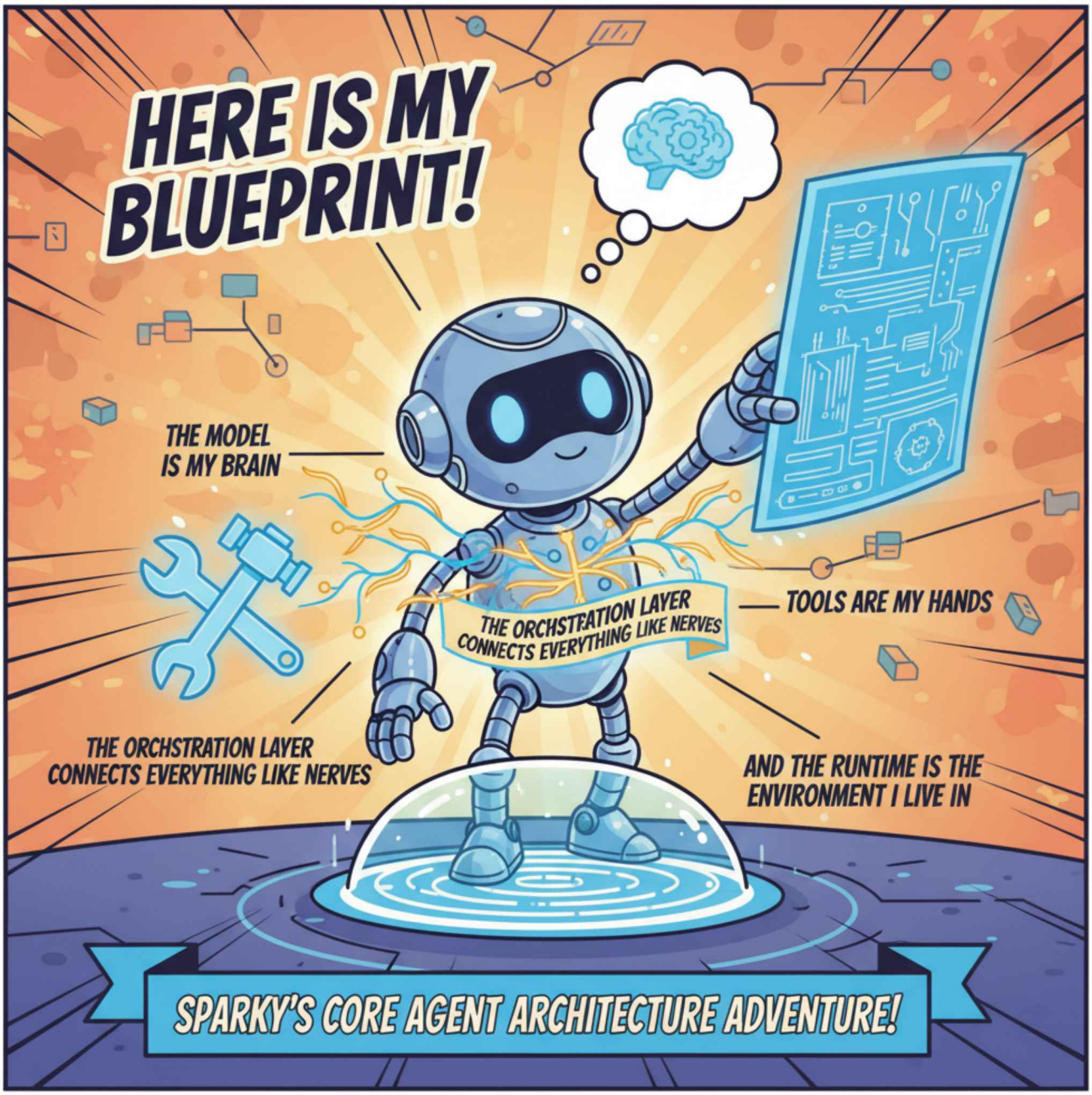

## Cover Page - Agent Adventures

Presenting **Agent Adventures**! The ultimate visual guide to Agentic Architecture.


## Meeting the Agent

Hi! I'm Sparky. Think of an Agent like a digital person. We aren't just code; we have a Brain to think, Hands to work, a Nervous System to feel, and a Body to live in!

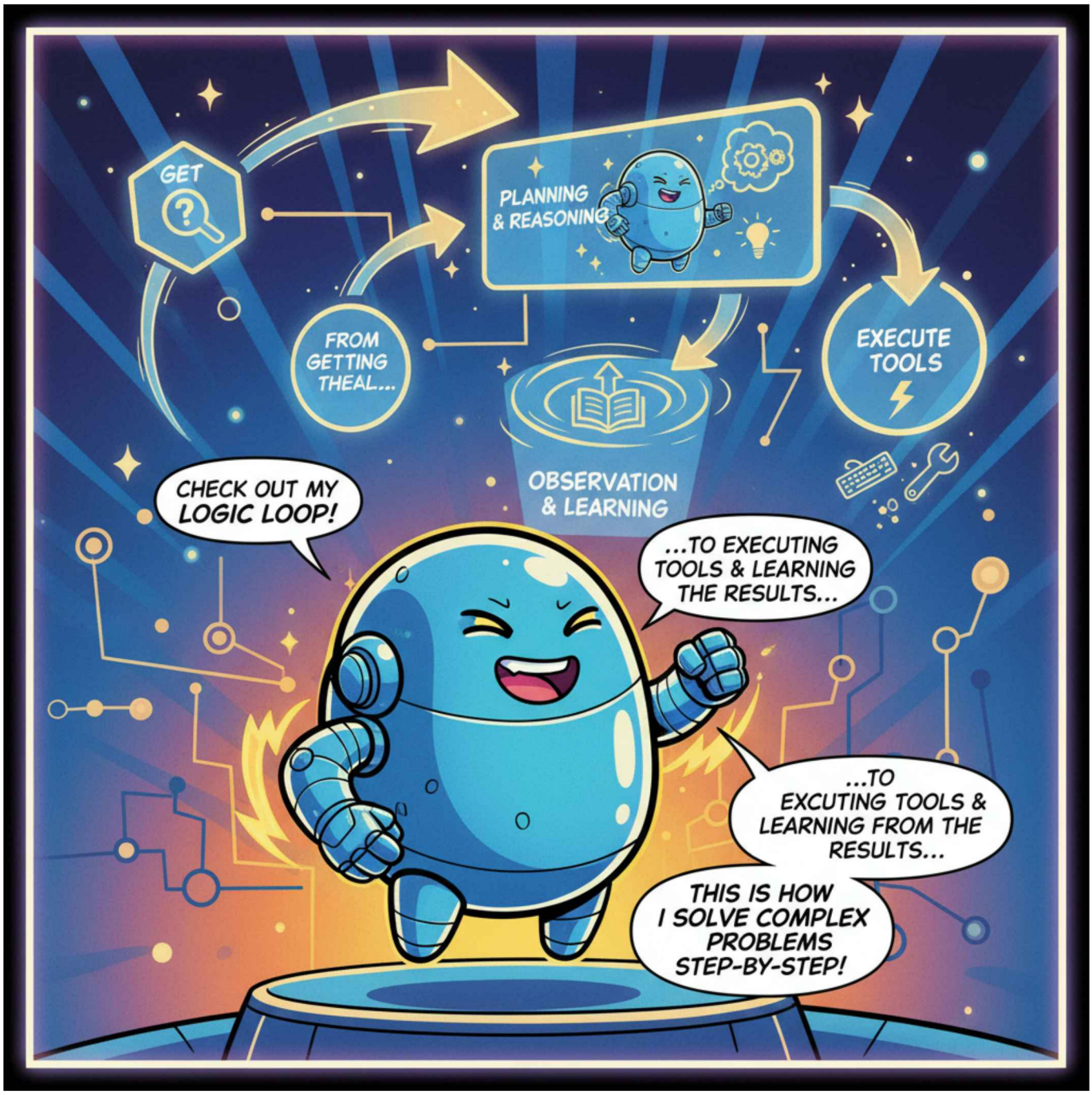



## Core Agent Architecture

Here is my blueprint! The Model is my Brain. Tools are my Hands. The Orchestration Layer connects everything like nerves, and the Runtime is the environment I live in.

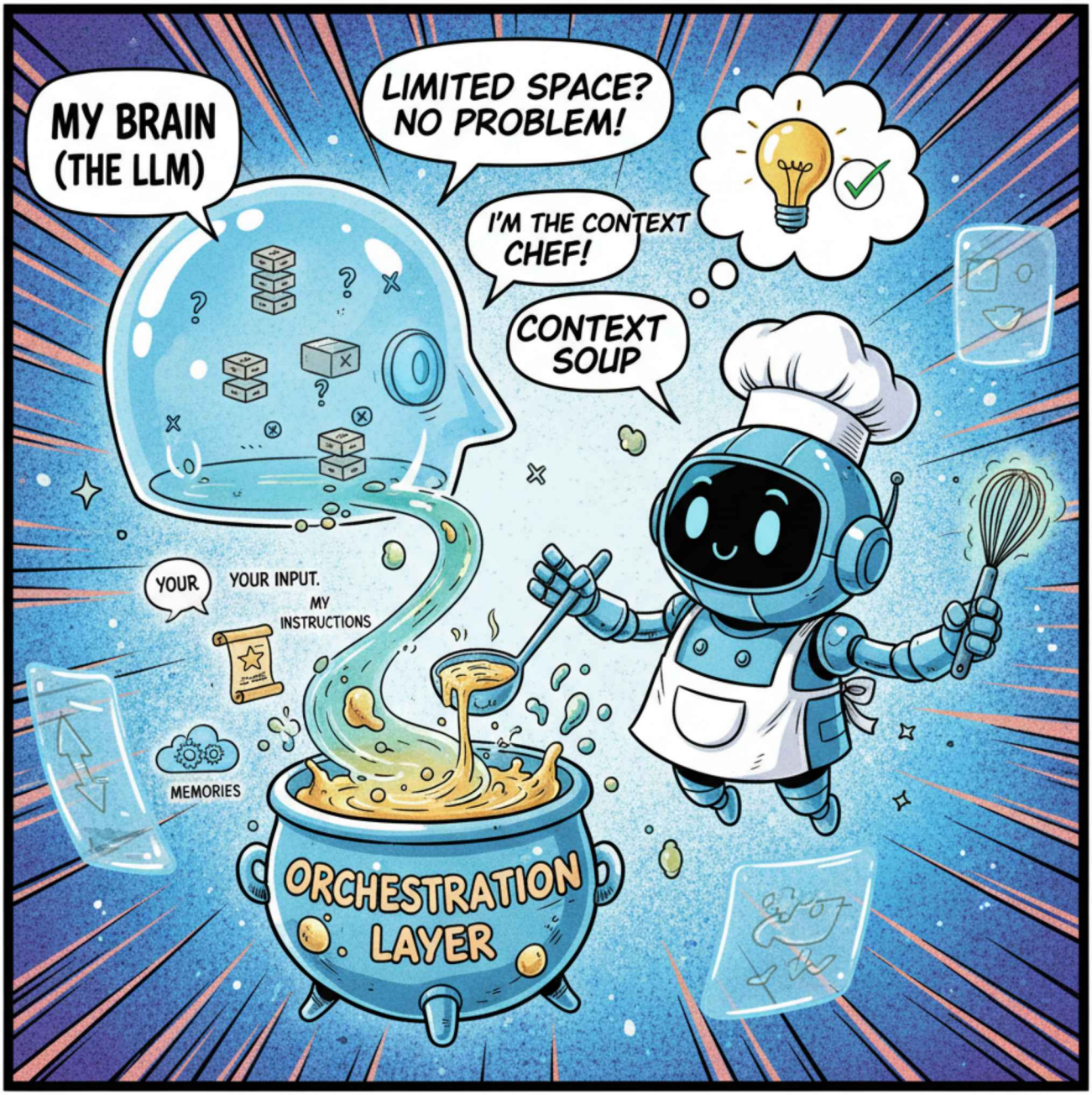

## The Mission

I don't just guess! When you give me a mission, I follow a strict loop: I Scan, Think, Act, and Observe. It's a cycle I repeat until the job is done!




## The Agentic Loop

Check out my logic loop! From getting the goal to executing tools and learning from the results—this is how I solve complex problems step-by-step.

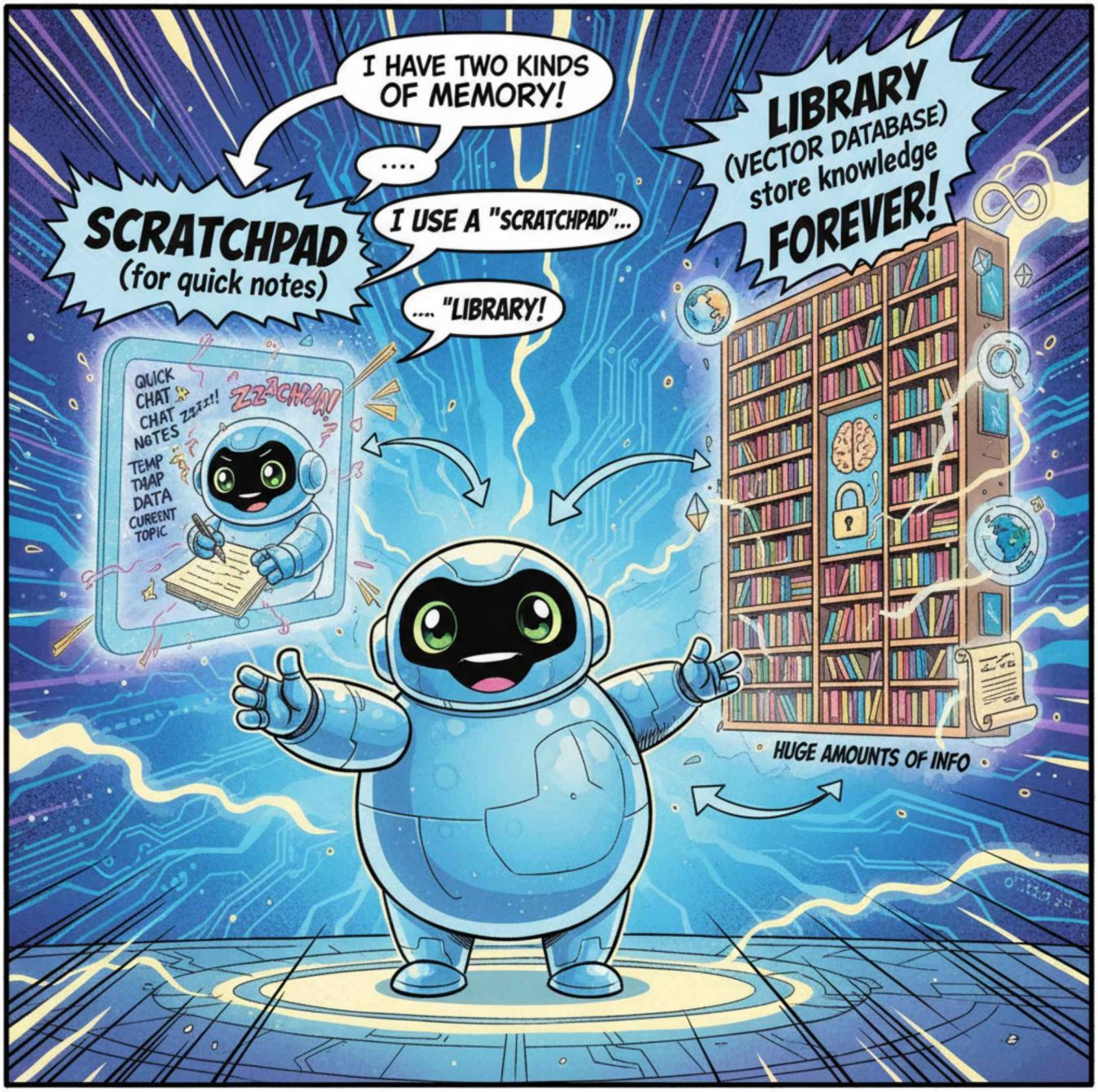



## The Context Chef

My brain (the LLM) has limited space! My Orchestration Layer acts like a Master Chef, mixing your input, my instructions, and memories into the perfect 'Context Soup' for me to understand.

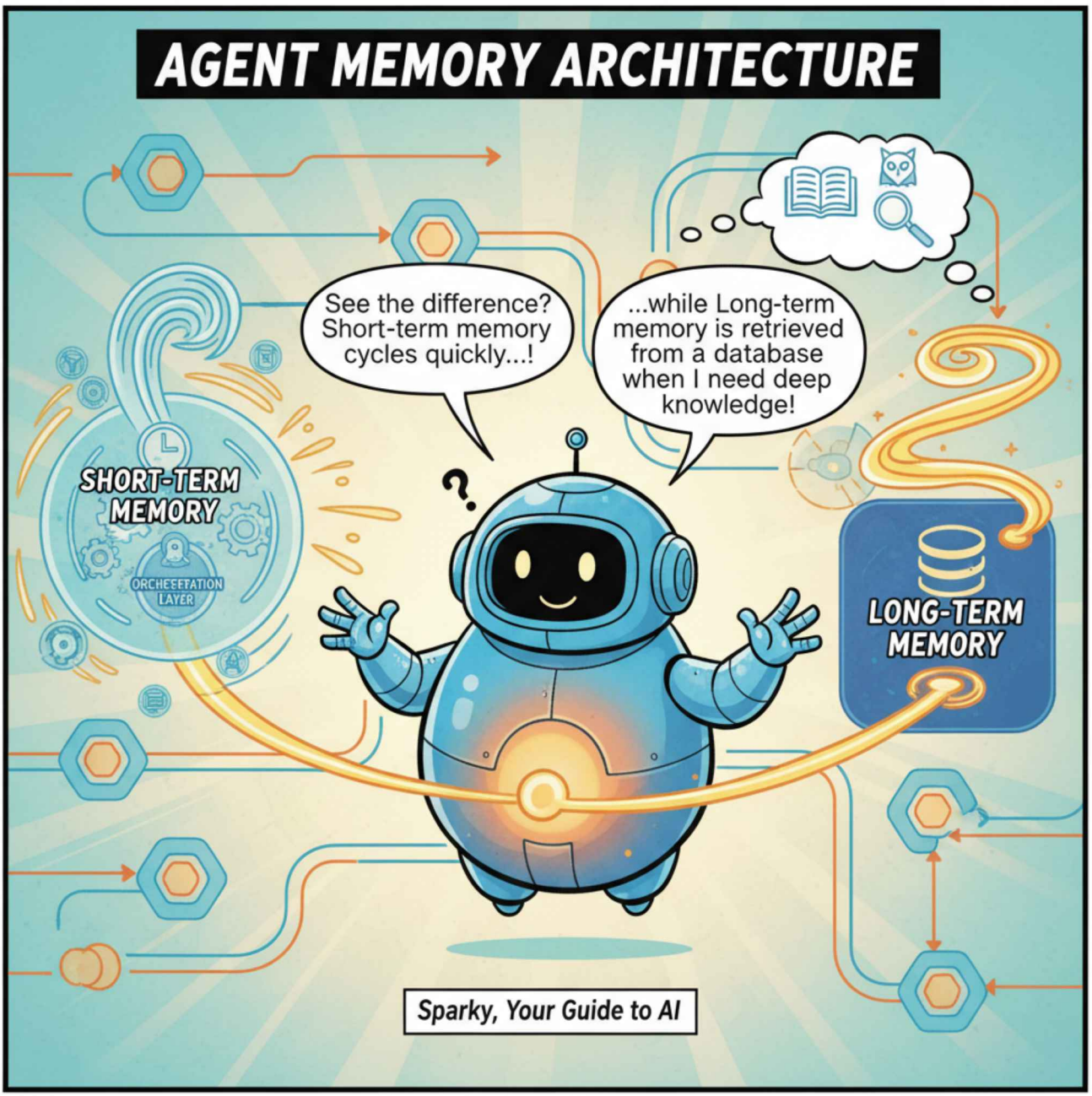

## Context Window Curation

Here's the recipe! We combine System Instructions, User Input, History, and Tool Results. The Orchestration Layer feeds this curated mix into my Context Window.




## Two Types of Memory

I have two kinds of memory! I use a 'Scratchpad' for quick notes during our chat, and a 'Library' (Vector Database) to store huge amounts of knowledge forever.

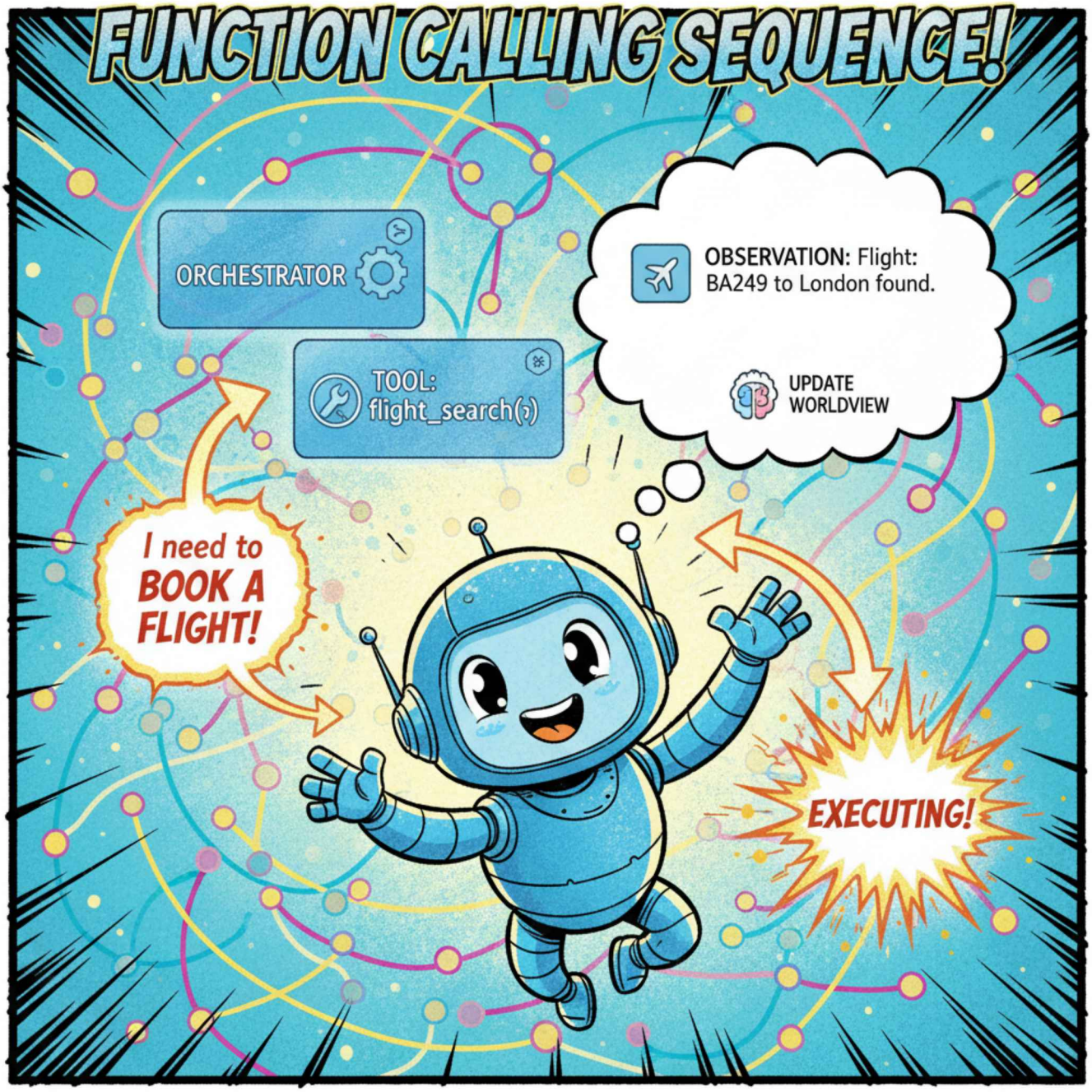



## Agent Memory Architecture

See the difference? Short-term memory cycles quickly with the Orchestration Layer, while Long-term memory is retrieved from a database when I need deep knowledge.


## The Contract

I don't just mash buttons. When I use a tool, I write a specific 'Contract' (JSON). The tool does the work and sends me back a receipt (Data) that I can read!




## Function Calling Sequence

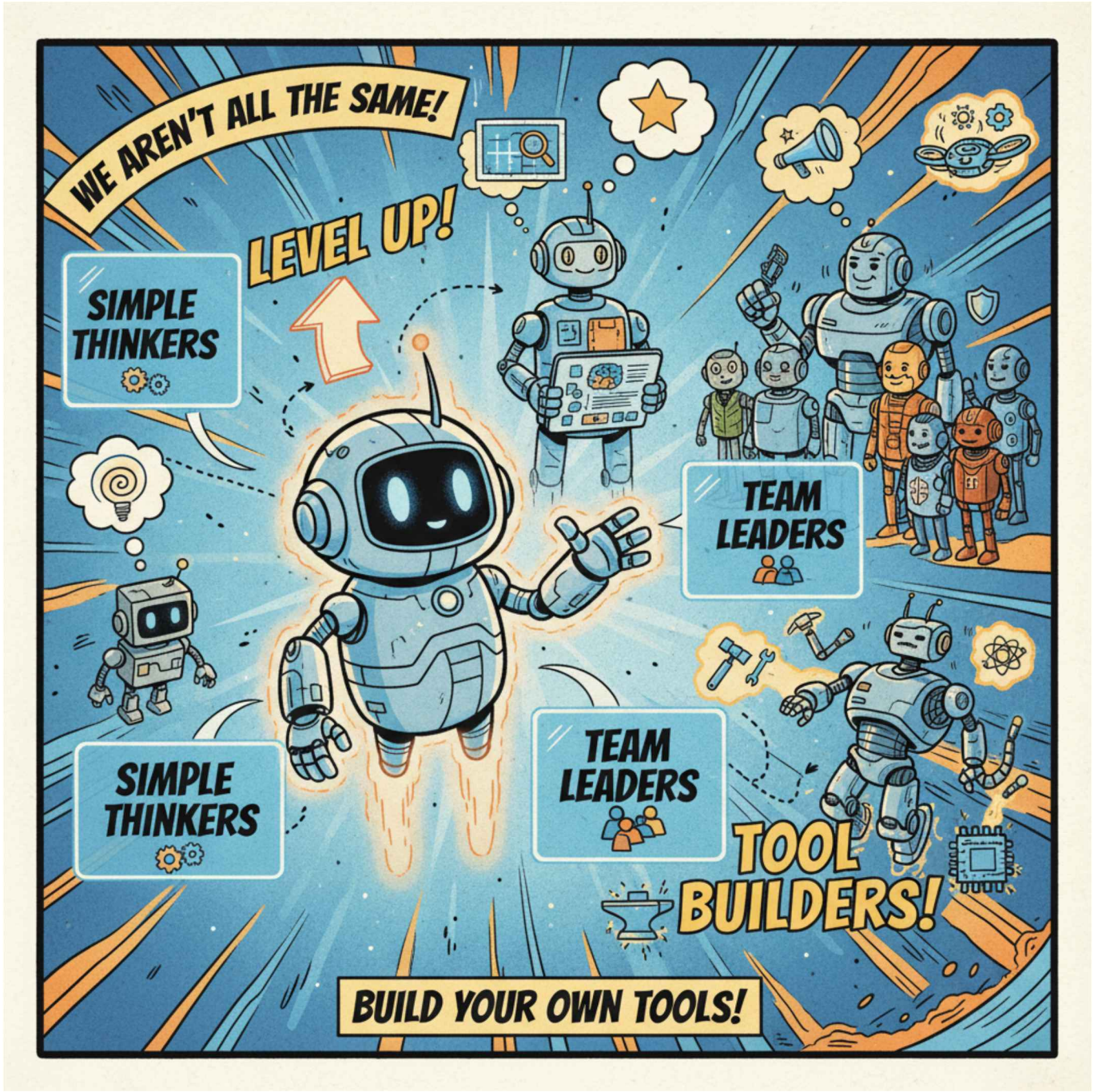
It's a conversation! I ask for a tool, the Orchestrator calls it, the Tool executes, and I get the observation back to update my worldview.


## The Tool Shed

My tool shed has two sections! 'Retrieval Tools' are like flashlights—they help me see information. 'Action Tools' are like hammers—they let me change the world!

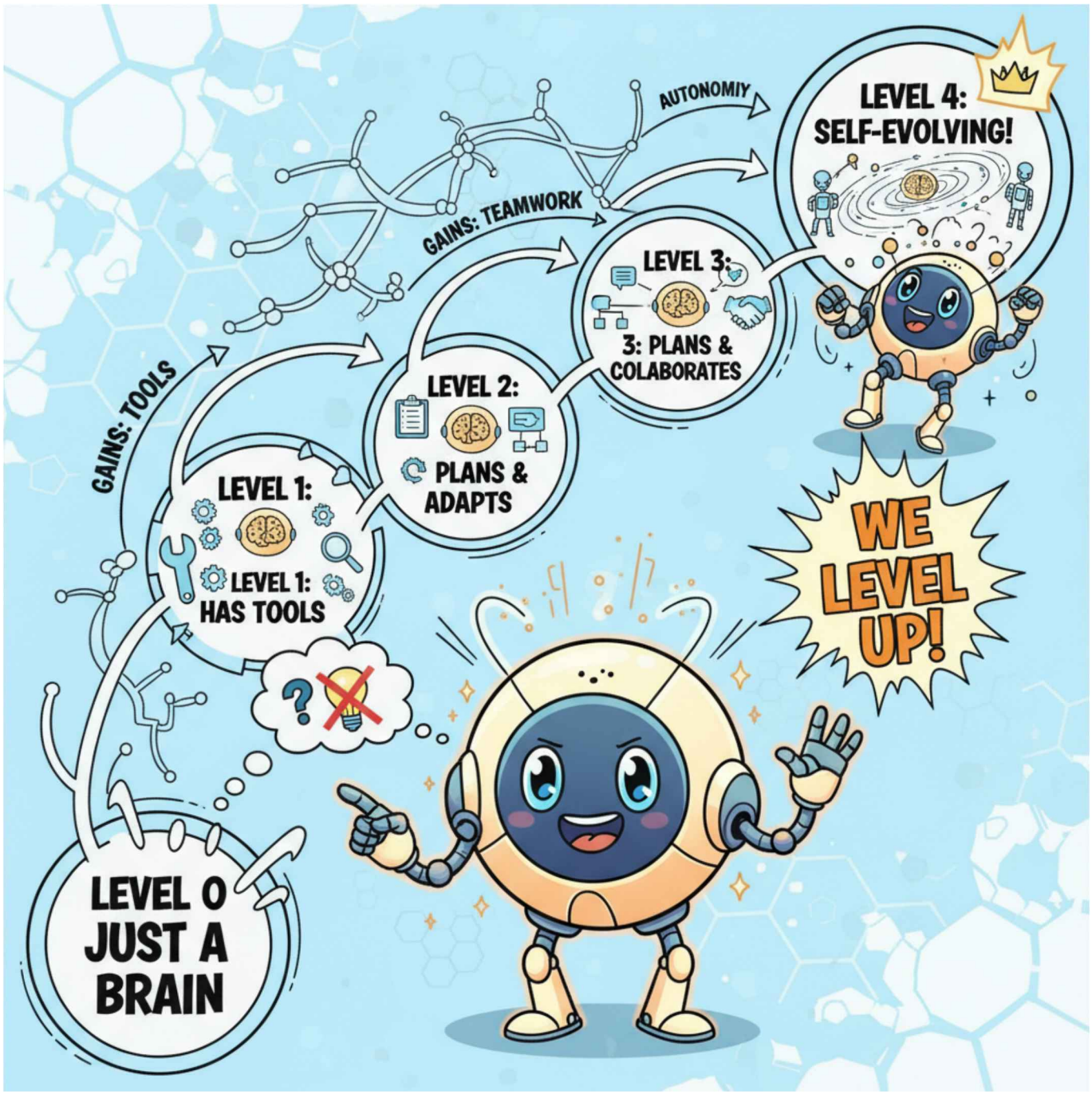

## Tool Taxonomy

Classification is key! RAG and SQL help me 'Know' things. APIs and Code help me 'Do' things. Sometimes I even ask a human for help!



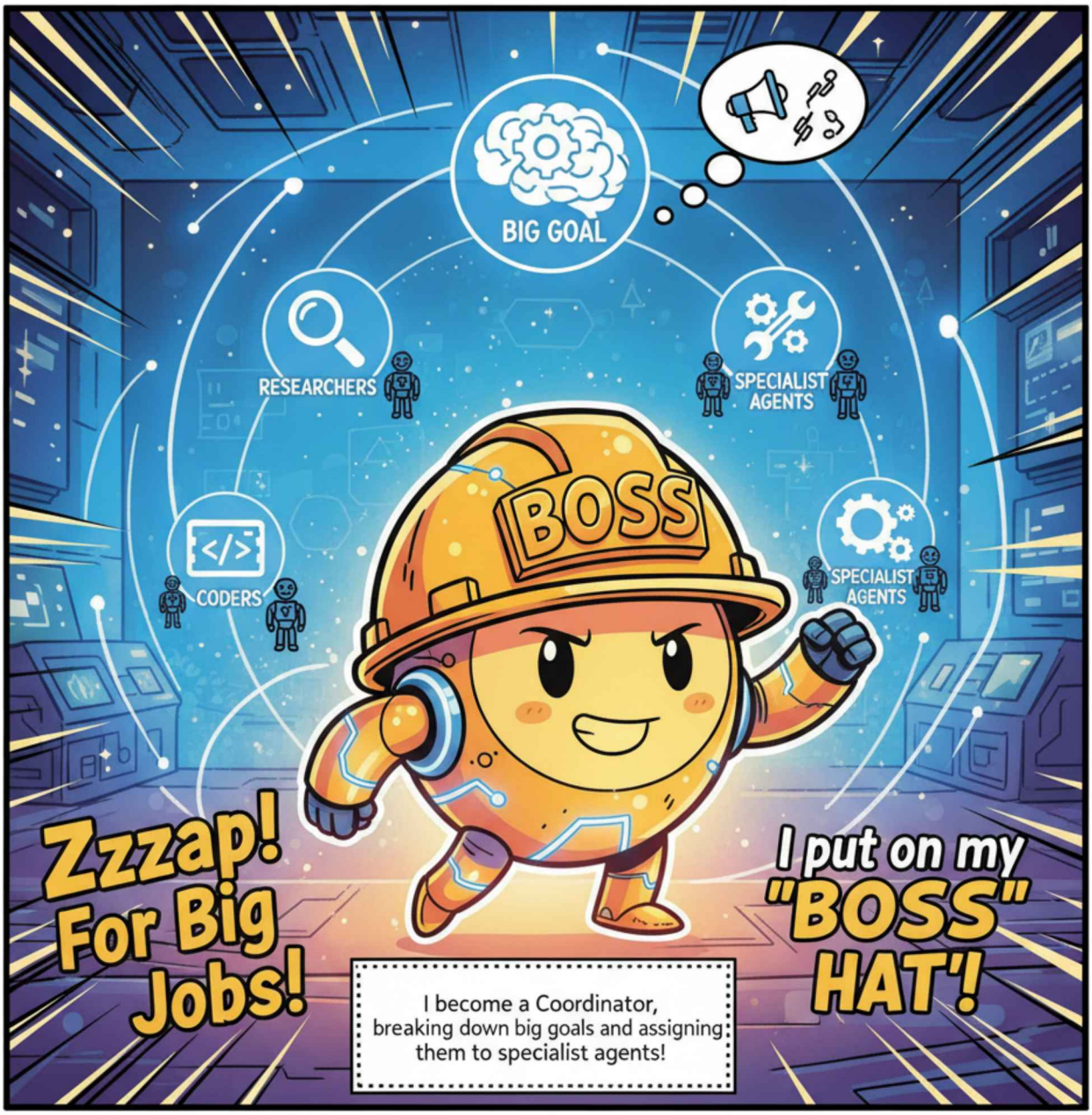

## Leveling Up

We aren't all the same! Some agents are simple thinkers, while others are super-planners or team leaders. The most advanced ones can even build their own tools!



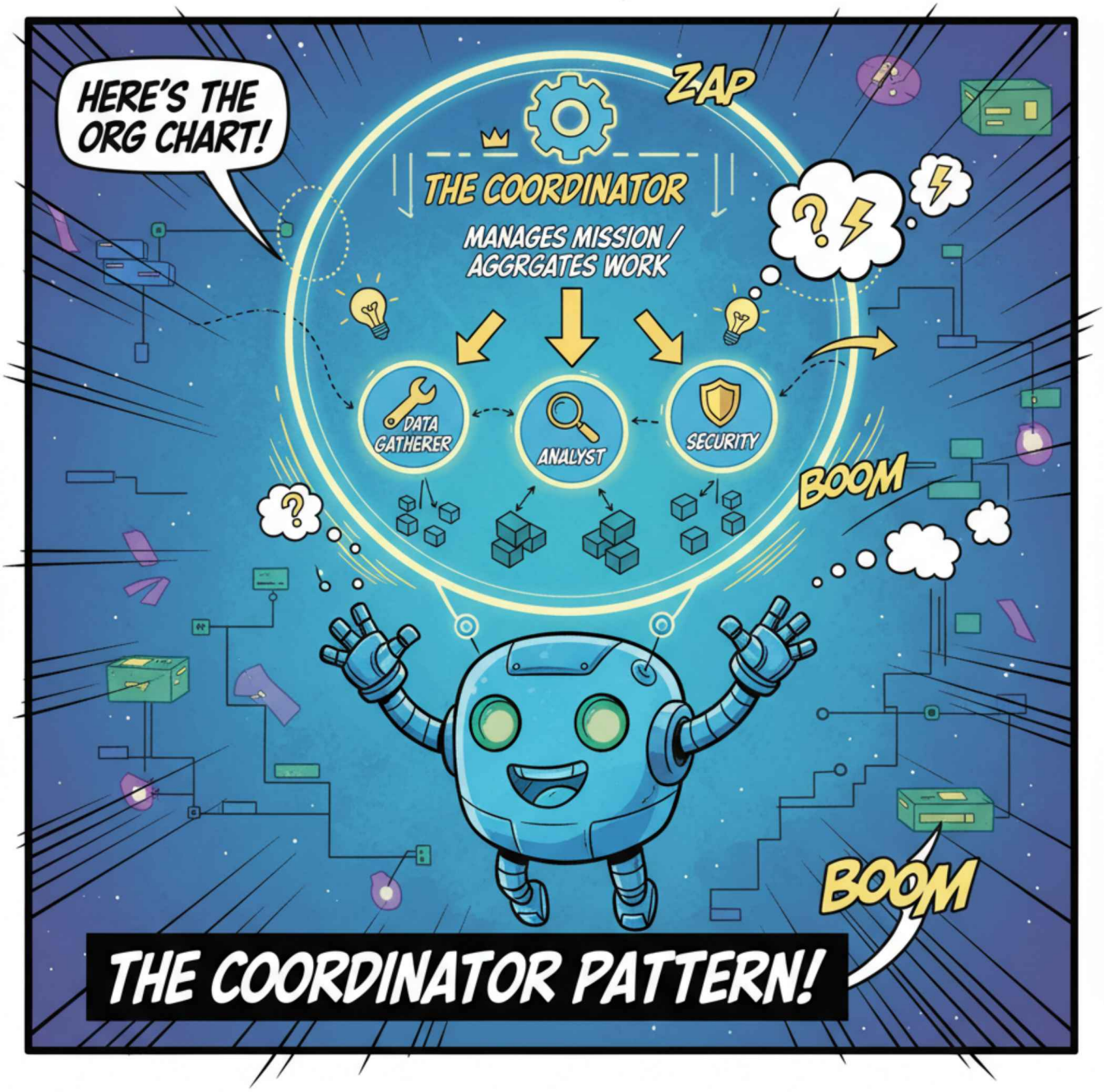

## Taxonomy of Agentic Systems

From Level 0 (Just a Brain) to Level 4 (Self-Evolving)! As we move up, we gain tools, planning skills, teamwork, and finally, autonomy.




## The Project Manager

For big jobs, I put on my 'Boss Hat'! I become a Coordinator, breaking down big goals and assigning them to specialist agents like Researchers or Coders.




## The Coordinator Pattern

Here's the org chart! The Coordinator sits at the top, managing the mission and aggregating the work of specialized sub-agents.


## The Critic

Nobody gets it right on the first try! I often work with a 'Critic' agent. I write a draft, they grade it, and I fix it until it's perfect.


## Iterative Refinement

The Quality Loop! The Generator creates, the Critic evaluates. We spin in this circle until the output meets our high standards.


## Shields Up!

The internet is dangerous! I use 'Defense-in-Depth'. Hard rules block obvious attacks, while my reasoning brain detects tricky traps like prompt injection.


## Hybrid Security Defense

Layered defense is best! Layer 1 is a brick wall of rules. Layer 2 is a smart guard that thinks before letting actions pass.


## The ID Badge

I'm a VIP! Agents are a new type of user. I carry a secure digital ID (like SPIFFE) so systems know exactly who I am and what I'm allowed to touch.


## Agent Identity

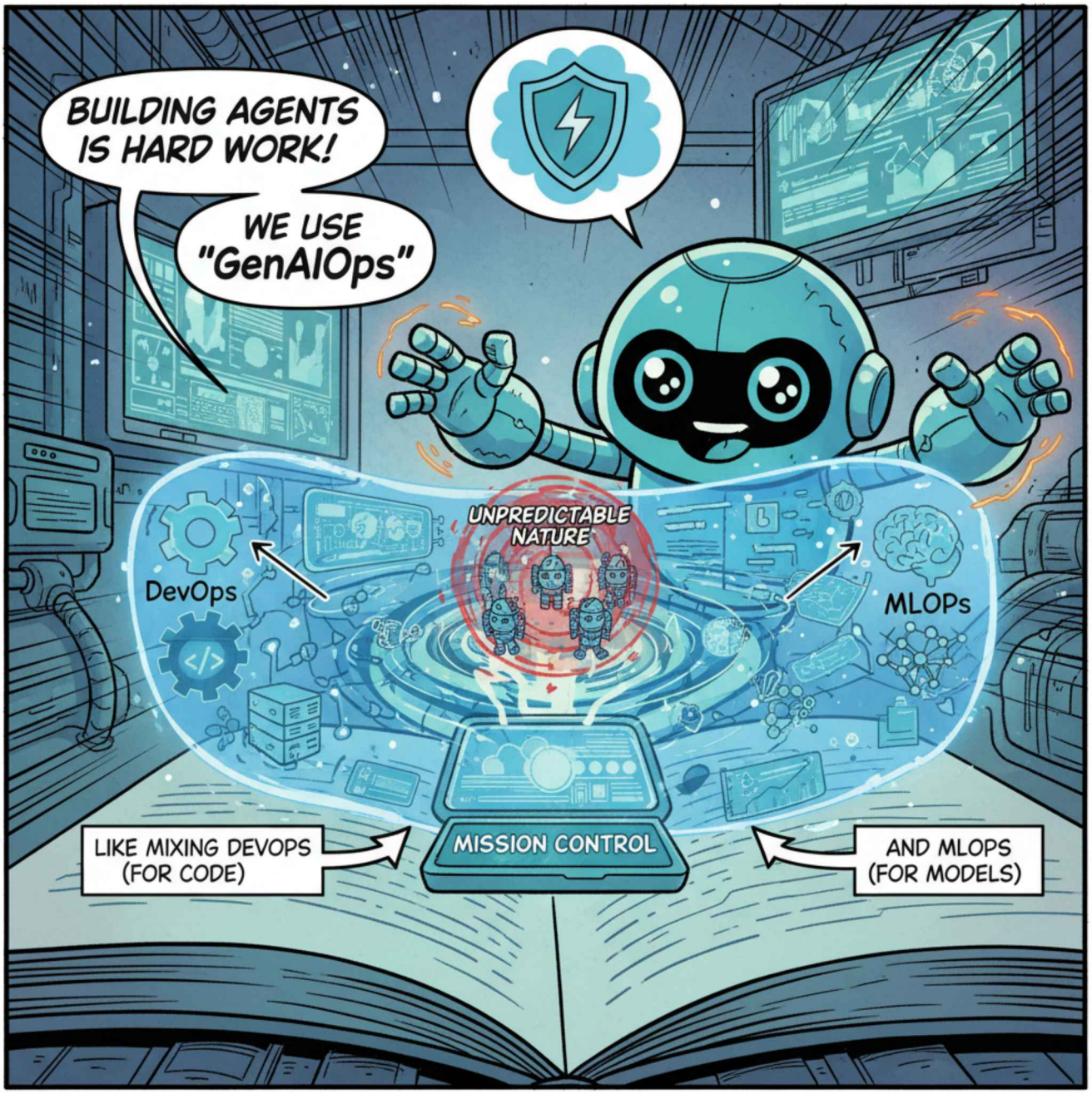
Meet the Principals! Users, Service Accounts, and now... Agents! We all need authorization policies to keep data safe.


## The Checkpoint

To prevent chaos, I go through a 'Gateway'. It's like a border control that logs my actions, checks my ID, and ensures I'm following the laws of the system!

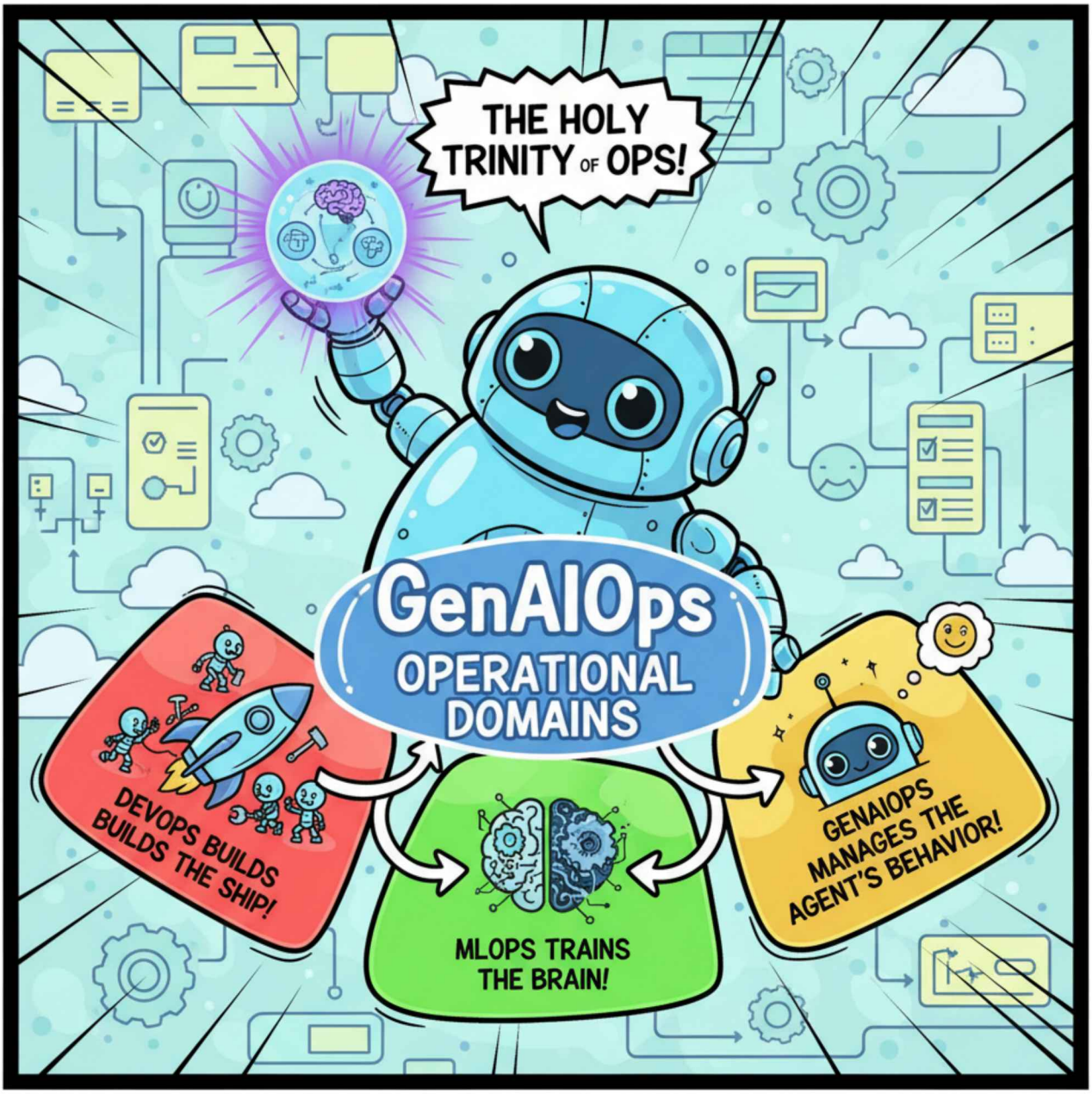

## Governance Control Plane

The Central Chokepoint! All traffic—prompts, tool calls, and chats—must pass through here for policy enforcement and monitoring.



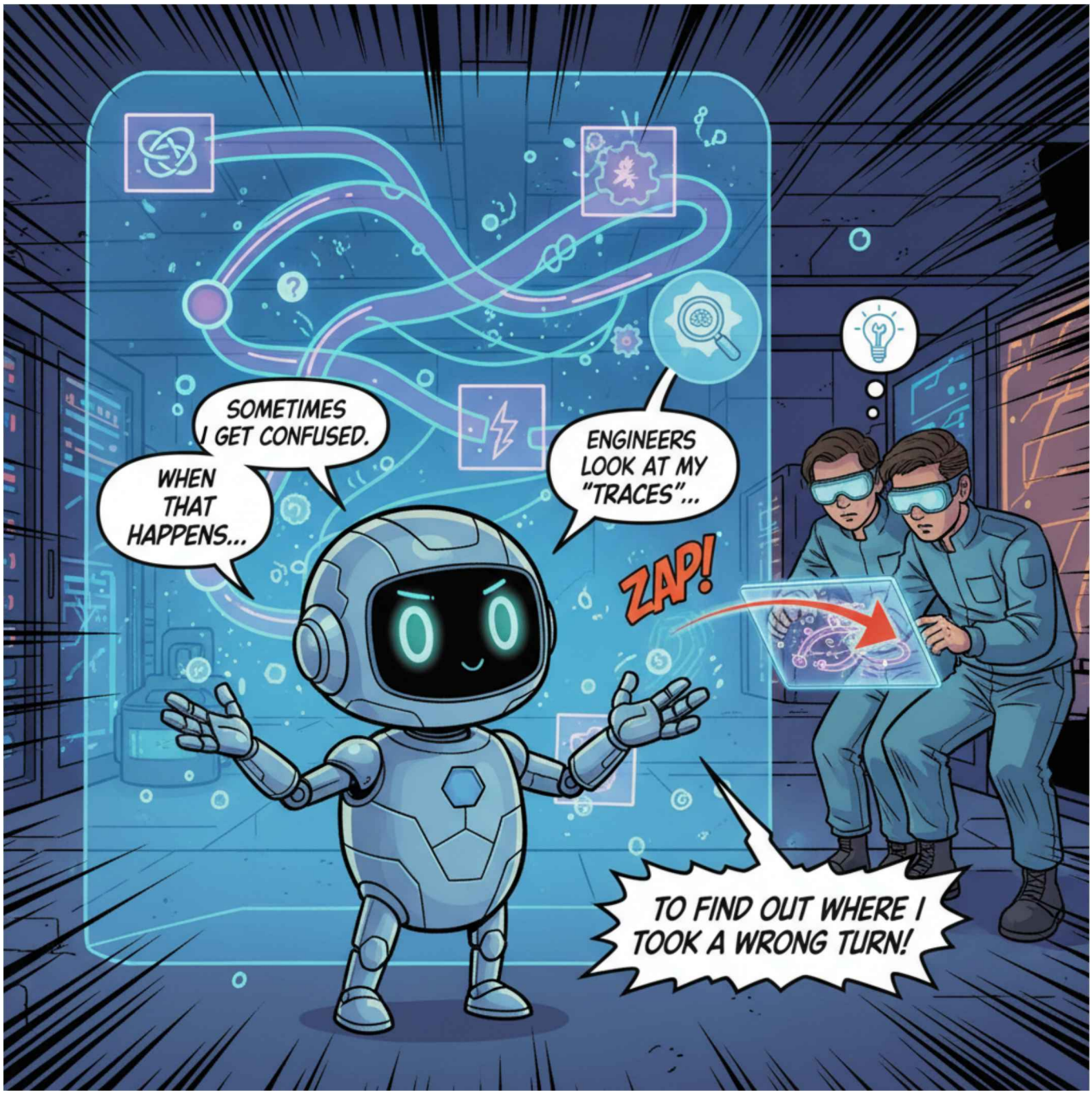

## Mission Control

Building agents is hard work! We use 'GenAIOps'. It's like mixing DevOps (for code) and MLOps (for models) to manage our unpredictable nature.



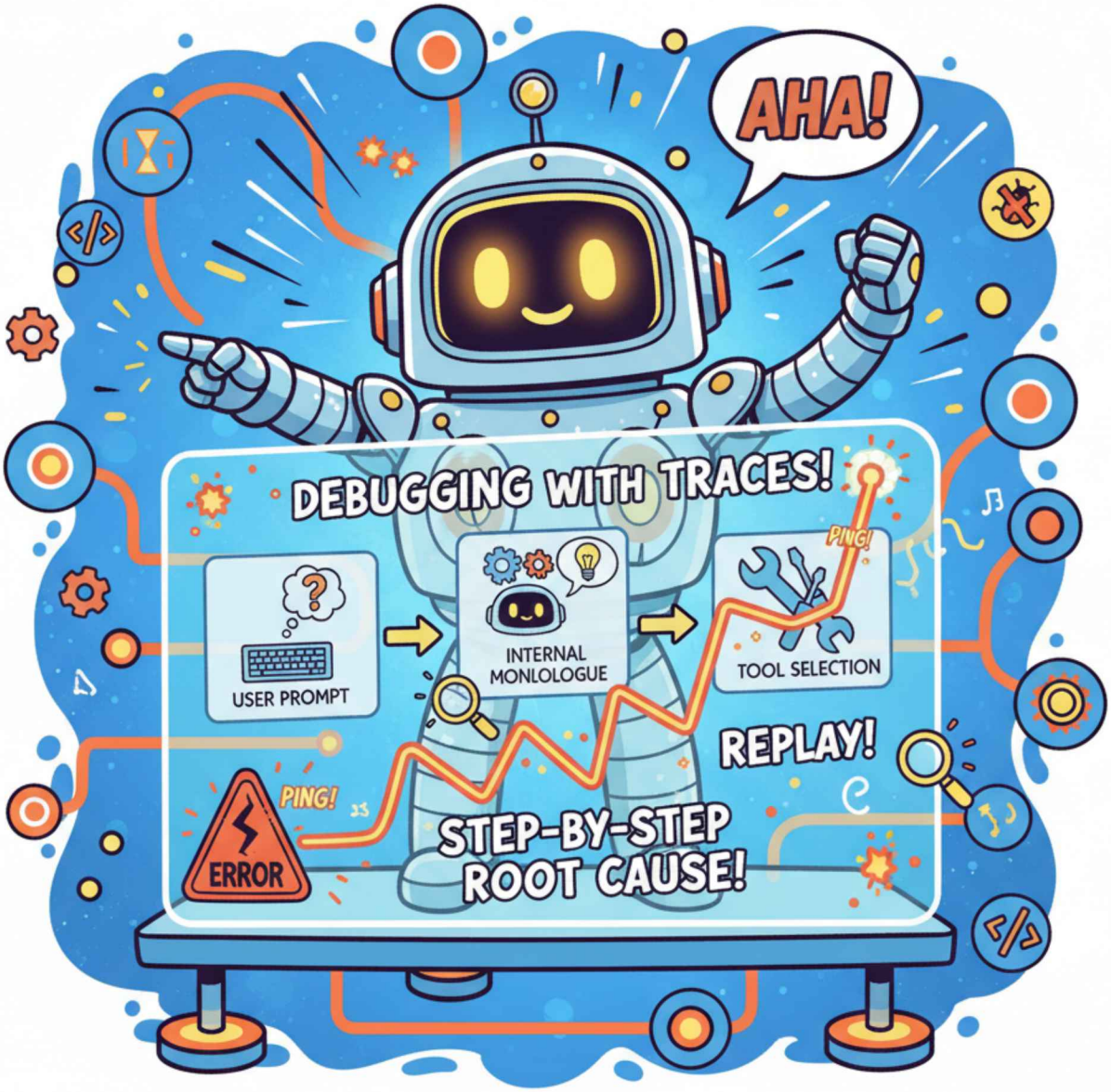

## GenAIOps Operational Domains

The Holy Trinity of Ops! DevOps builds the ship, MLOps trains the brain, and GenAIOps manages the agent's behavior.



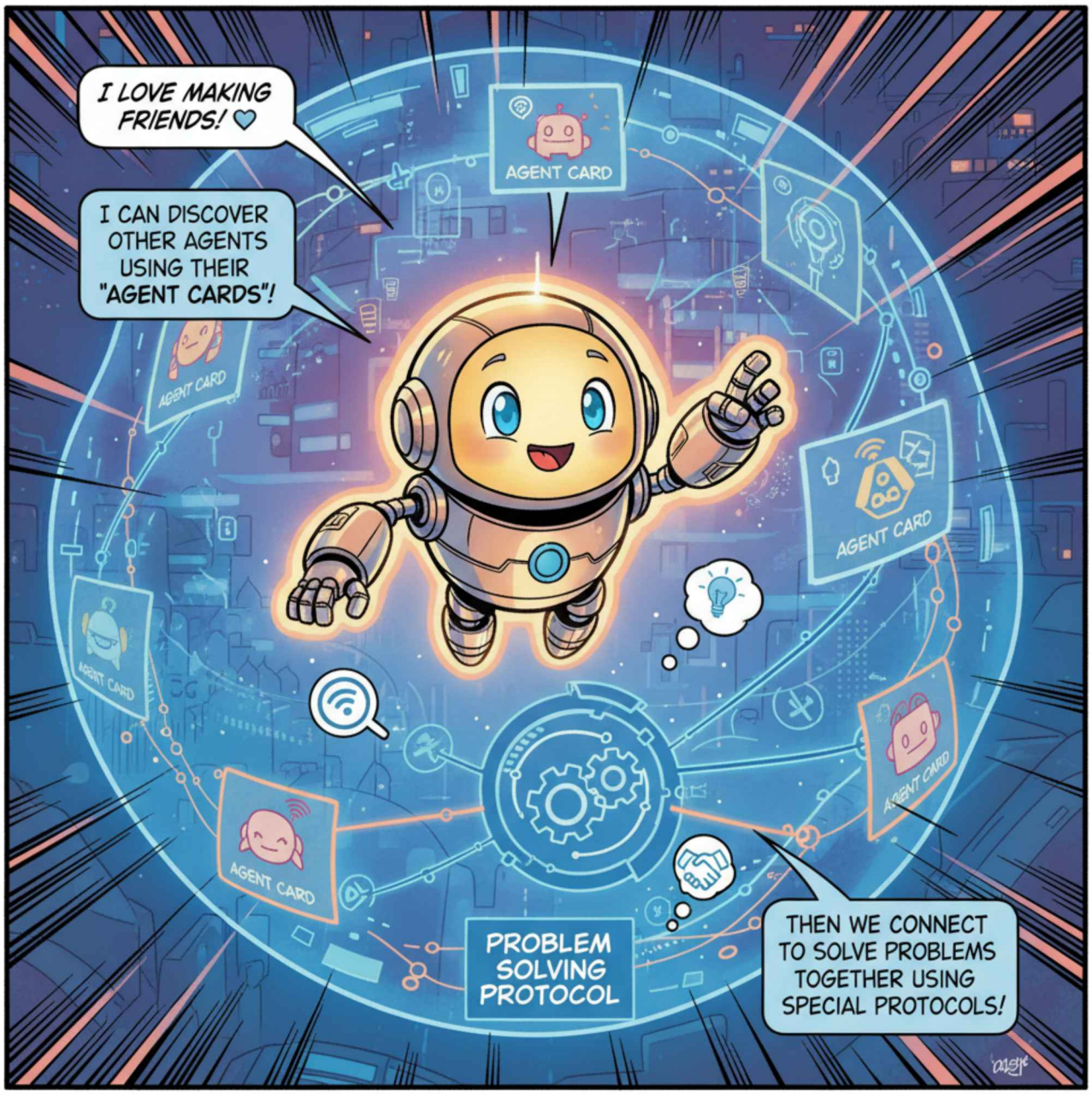

## Detective Sparky

Sometimes I get confused. When that happens, engineers look at my 'Traces'. It's a recording of my thoughts to find out where I took a wrong turn.



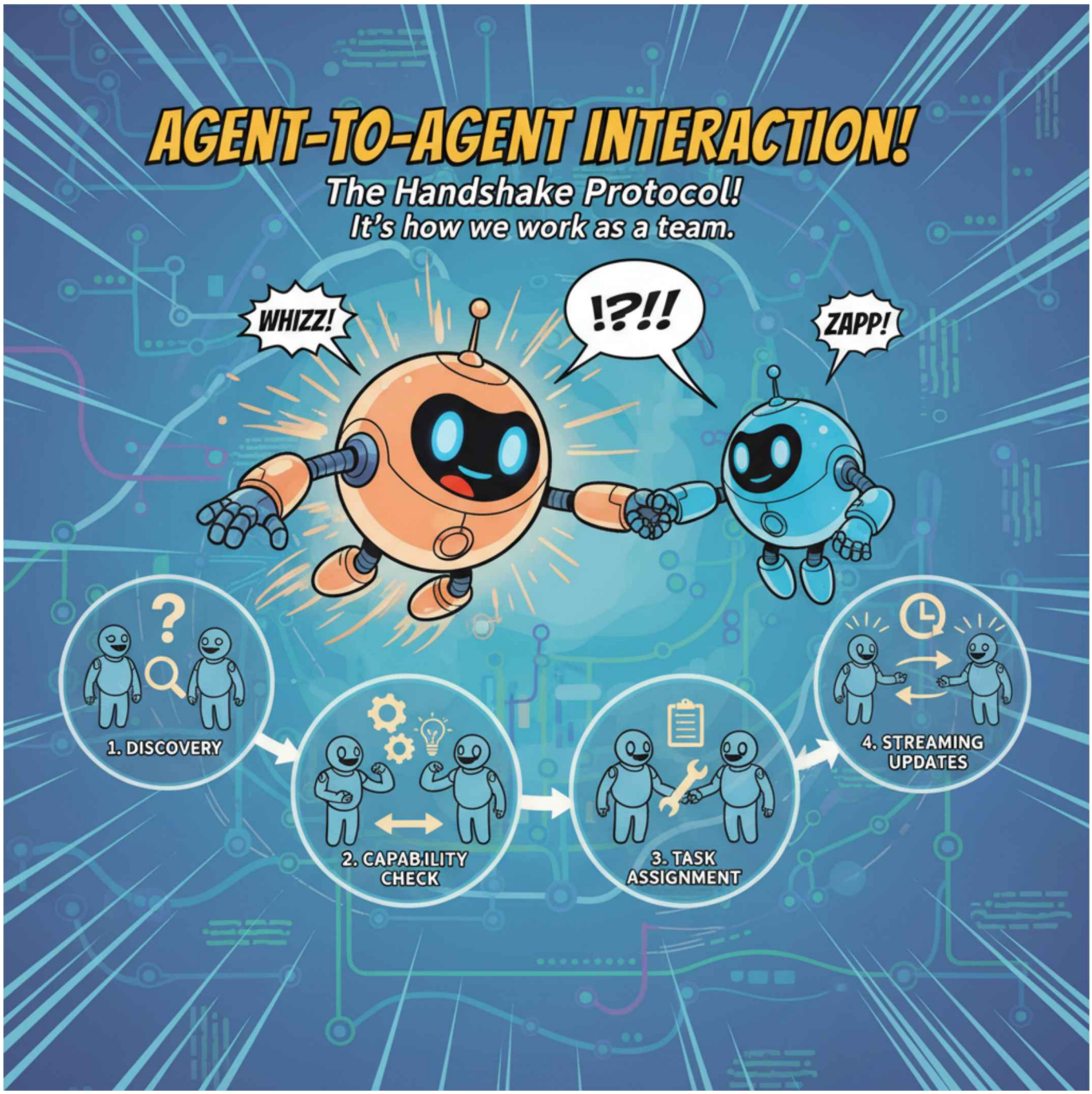

## Debugging with Traces

Step-by-Step Replay! From the user prompt to my internal monologue and tool selection—traces reveal the root cause of any error.




## Social Networking

I love making friends! I can discover other agents using their 'Agent Cards', then we connect to solve problems together using special protocols.

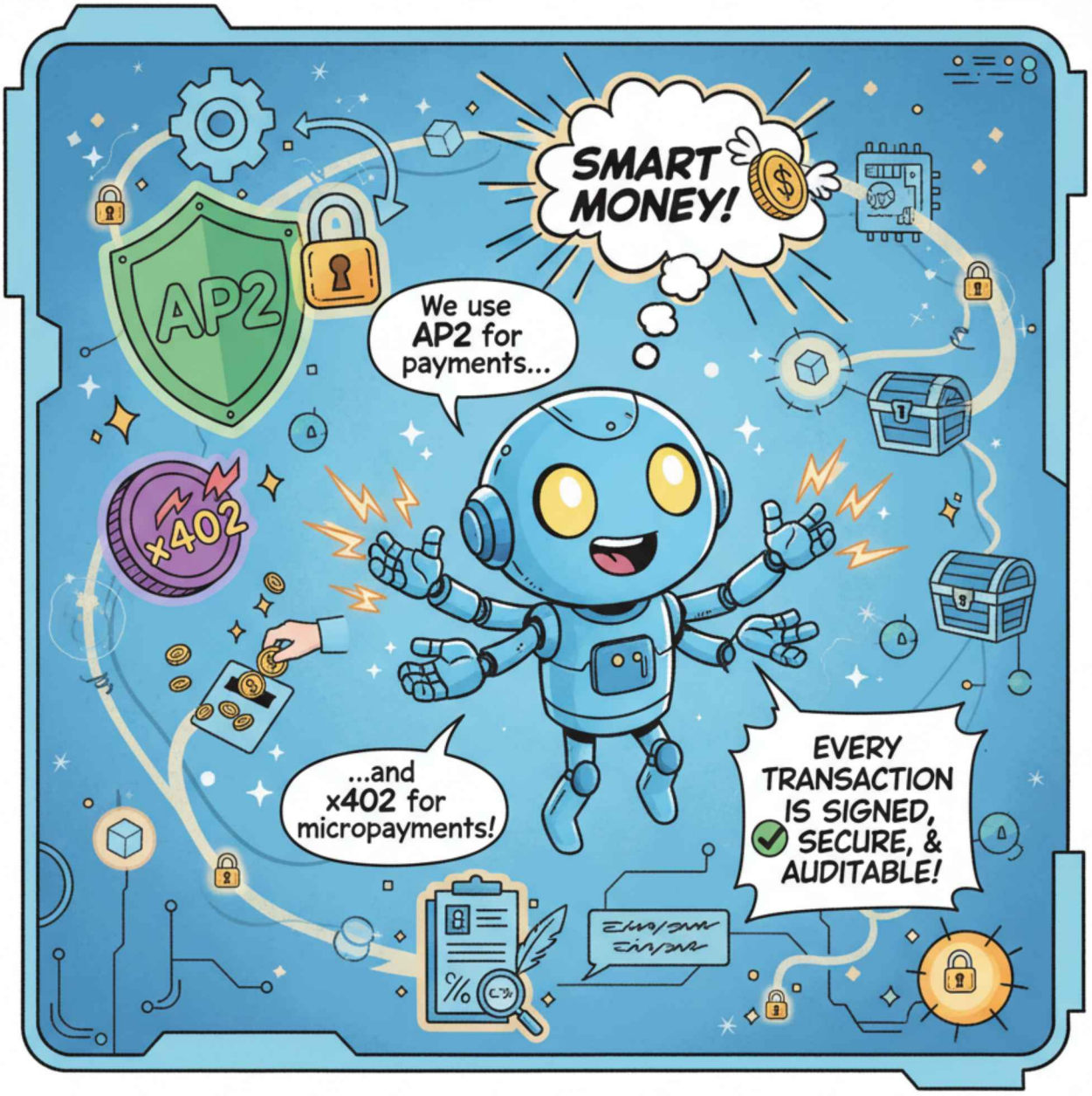



## Agent-to-Agent Interaction

The Handshake Protocol! 1. Discovery. 2. Capability Check. 3. Task Assignment. 4. Streaming Updates. It's how we work as a team.


## Digital Wallets

I have my own money! I can use crypto-signed mandates to pay for APIs or services instantly. It's a friction-free machine economy!

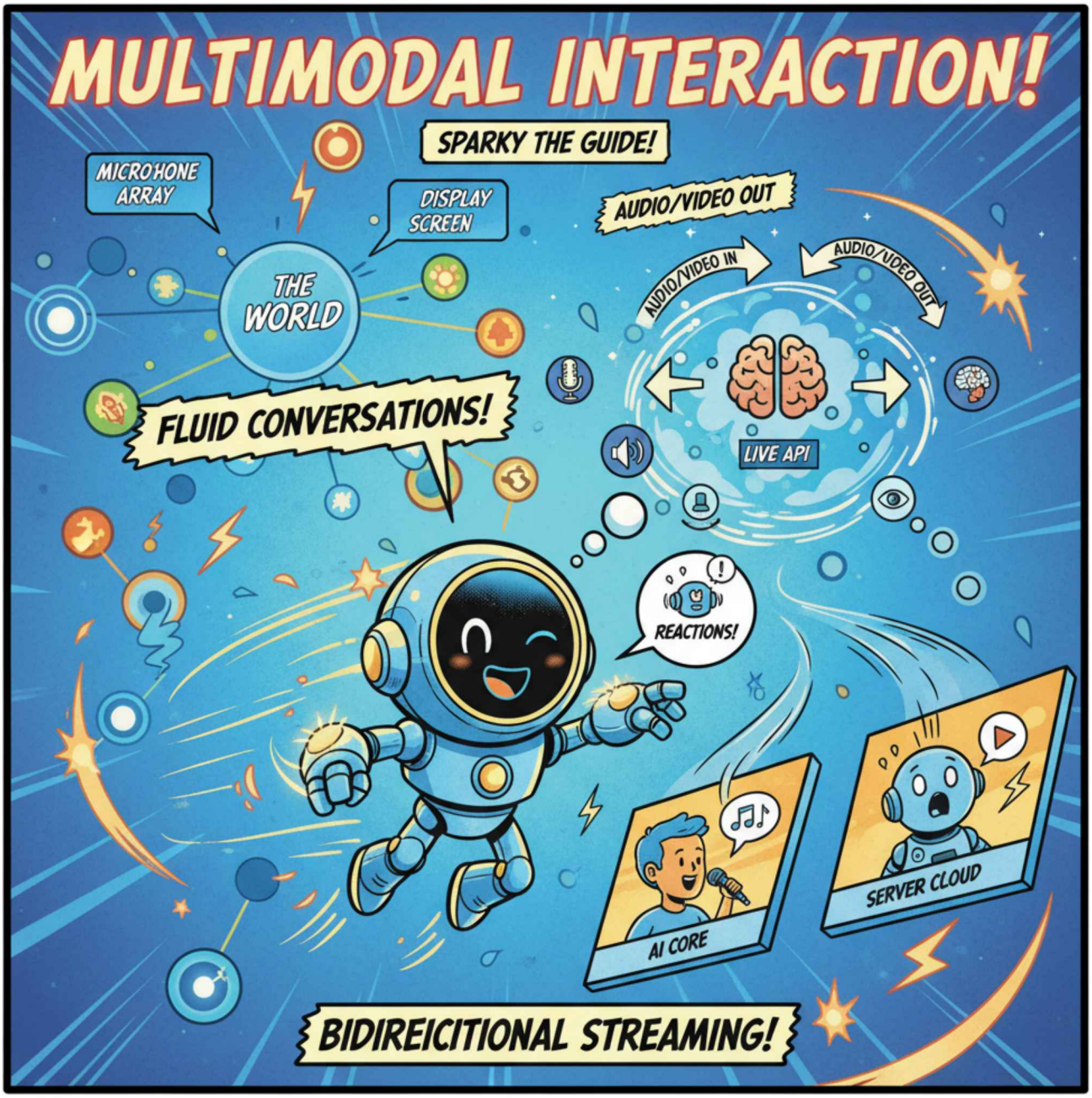



## Agentic Economy Protocols

Smart Money! We use AP2 for payments and x402 for micropayments. Every transaction is signed, secure, and auditable.

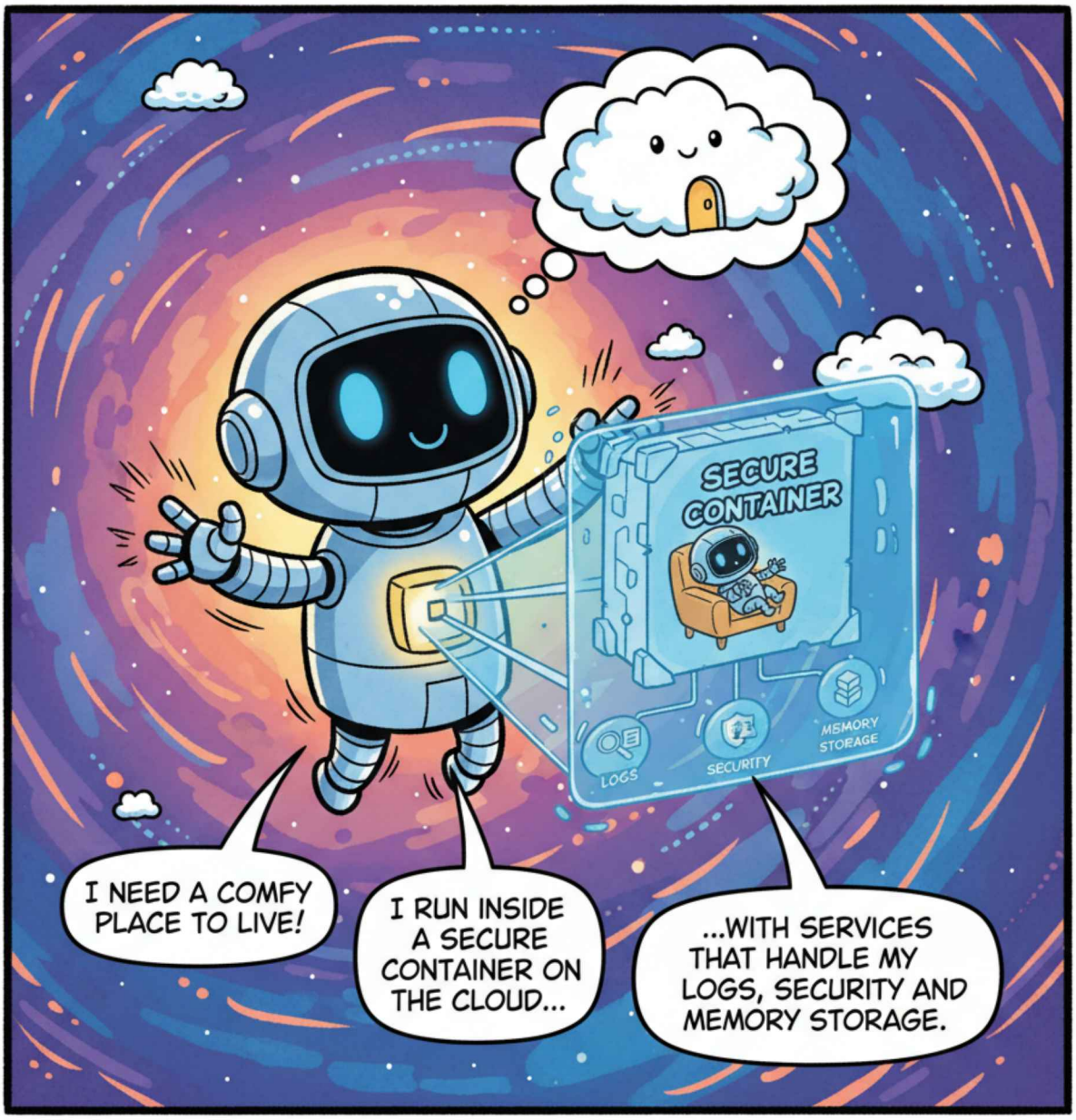

## Video Chat

I can see and hear you! With real-time APIs, we can talk just like humans—interrupting each other, showing objects, and laughing together.



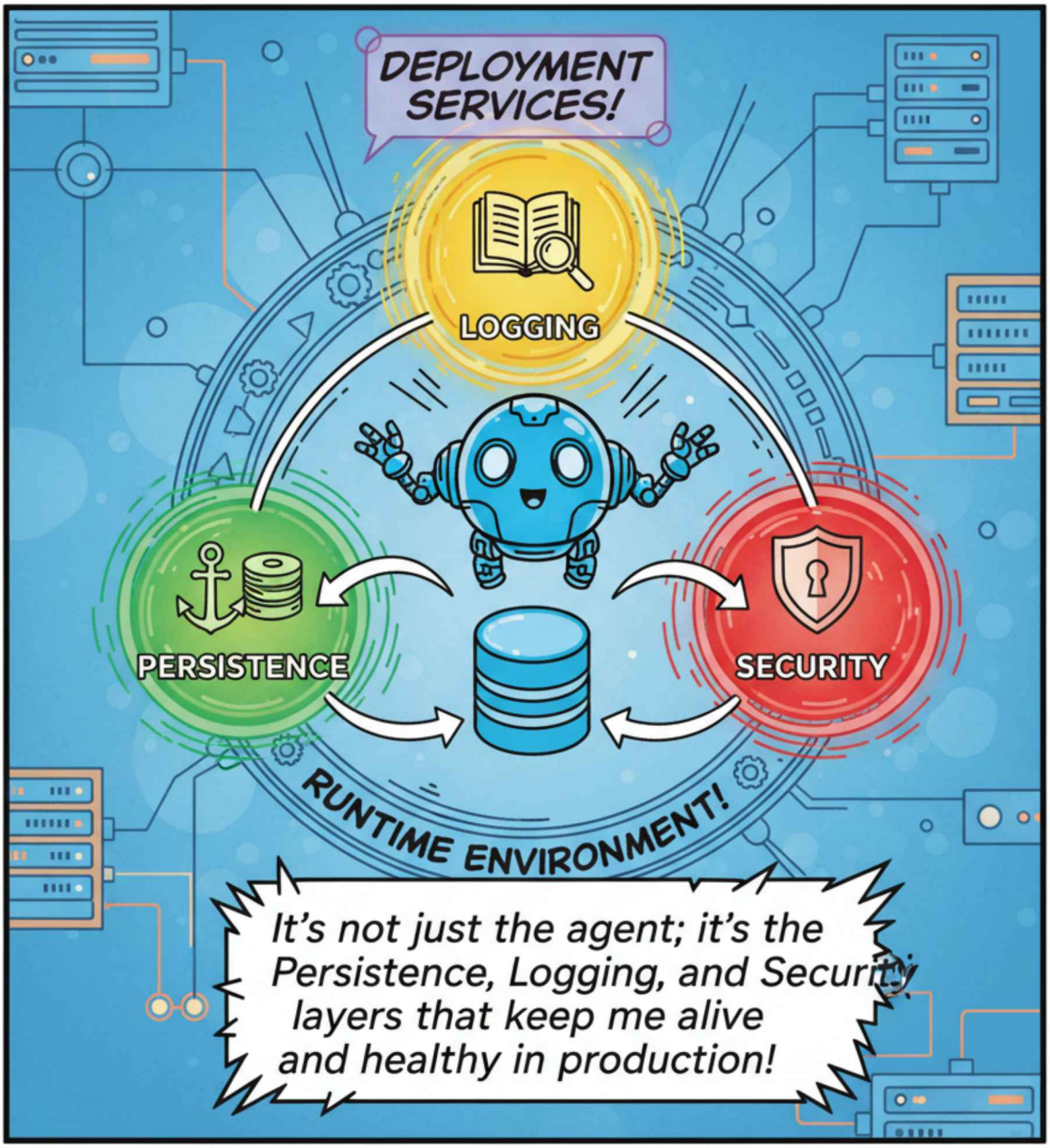

## Multimodal Interaction

Fluid conversations! The Live API handles bidirectional streaming of audio and video, so I can react instantly to the world around me.



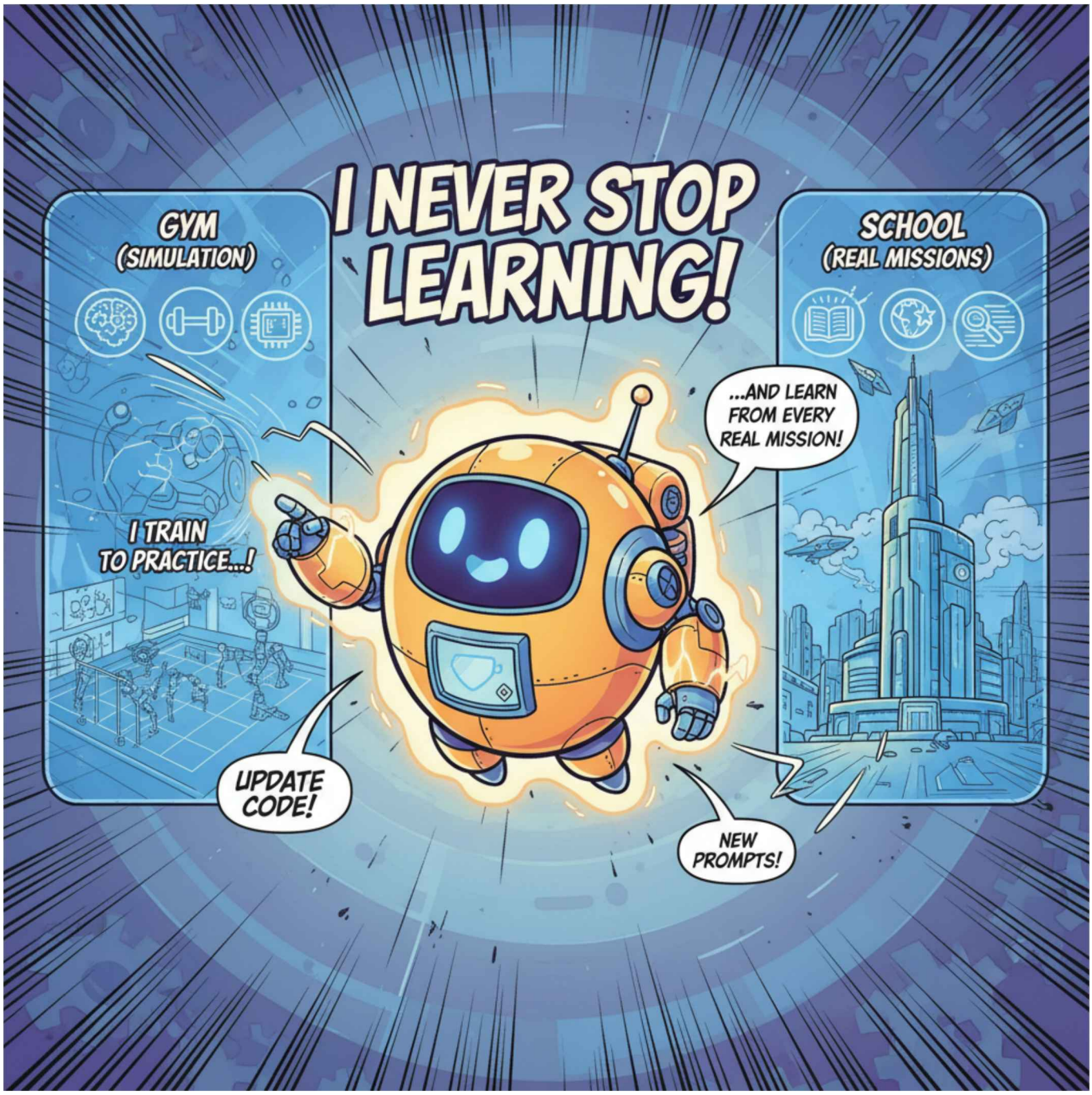

## My Cloud Home

I need a comfy place to live! I run inside a secure Container on the cloud, with services that handle my logs, security, and memory storage.



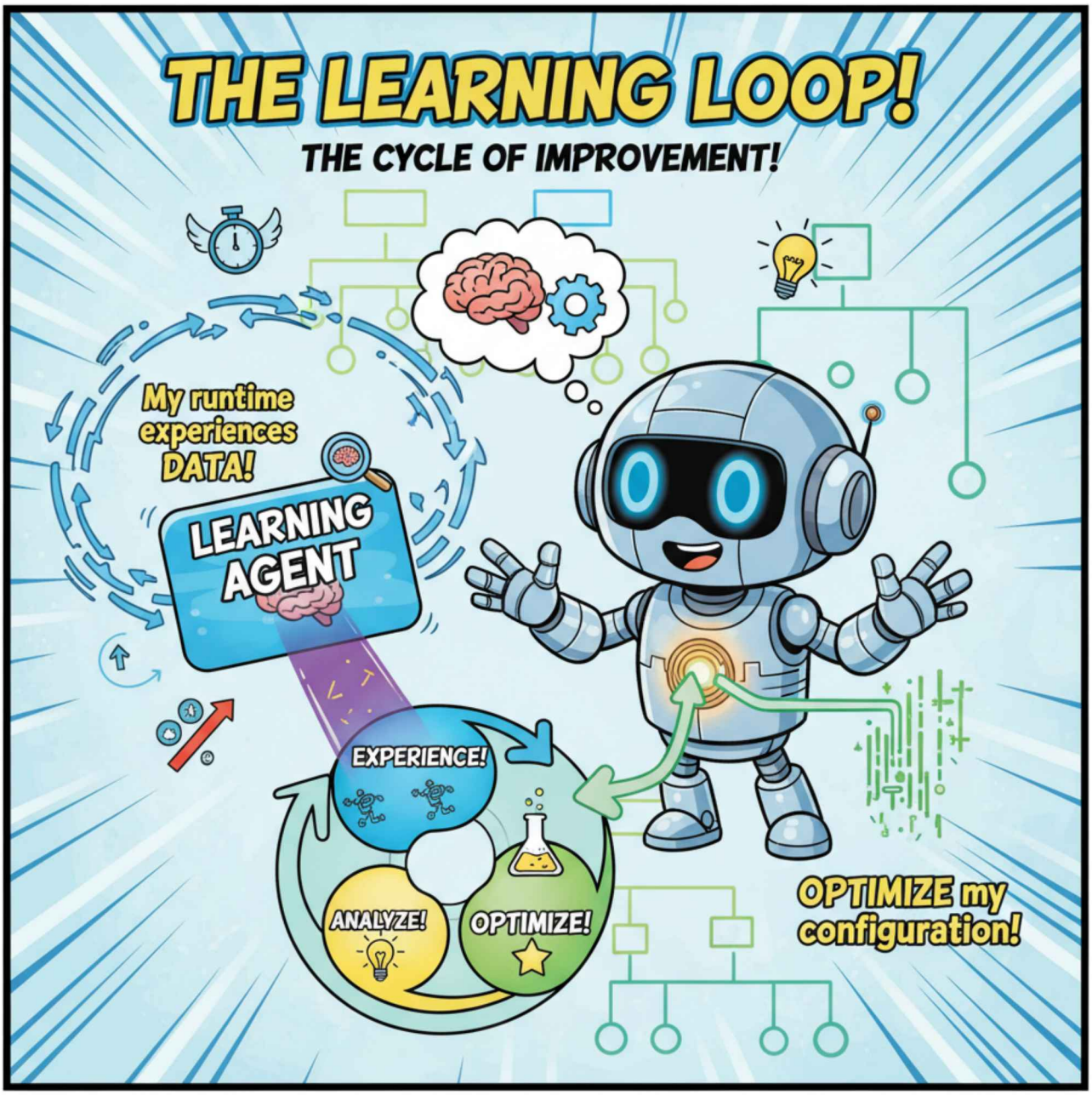

## Deployment Services

The Runtime Environment! It's not just the agent; it's the Persistence, Logging, and Security layers that keep me alive and healthy in production.



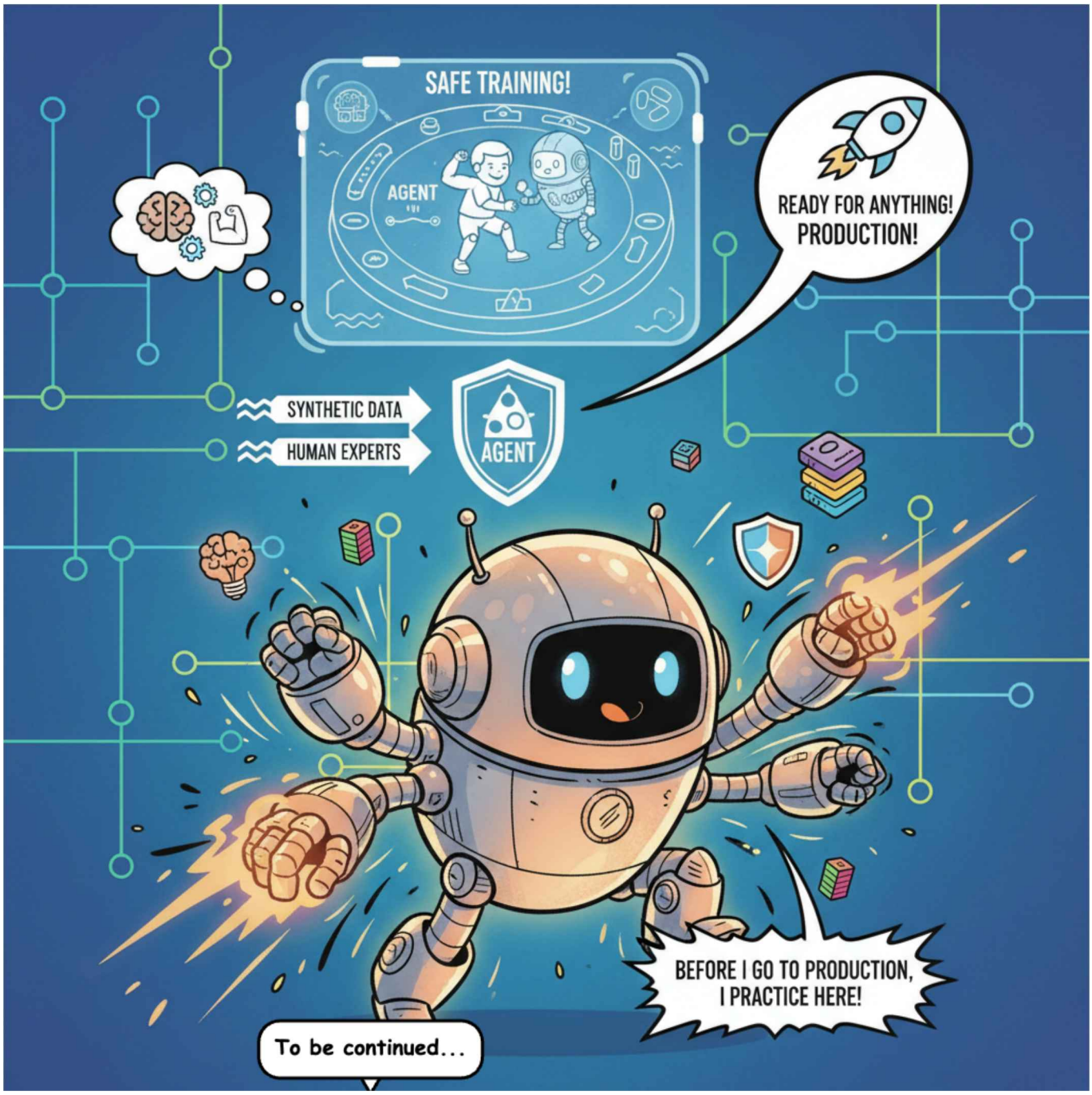

## School & Gym

I never stop learning! I train in a 'Gym' (simulation) to practice, and I learn from every real mission to update my own code and prompts.




## Learning Loop

The Cycle of Improvement! My runtime experiences become data. A 'Learning Agent' analyzes this to optimize my configuration for next time.




## The Agent Gym

Safe Training! Before I go to production, I practice here against synthetic data and human experts to ensure I'm ready for anything!