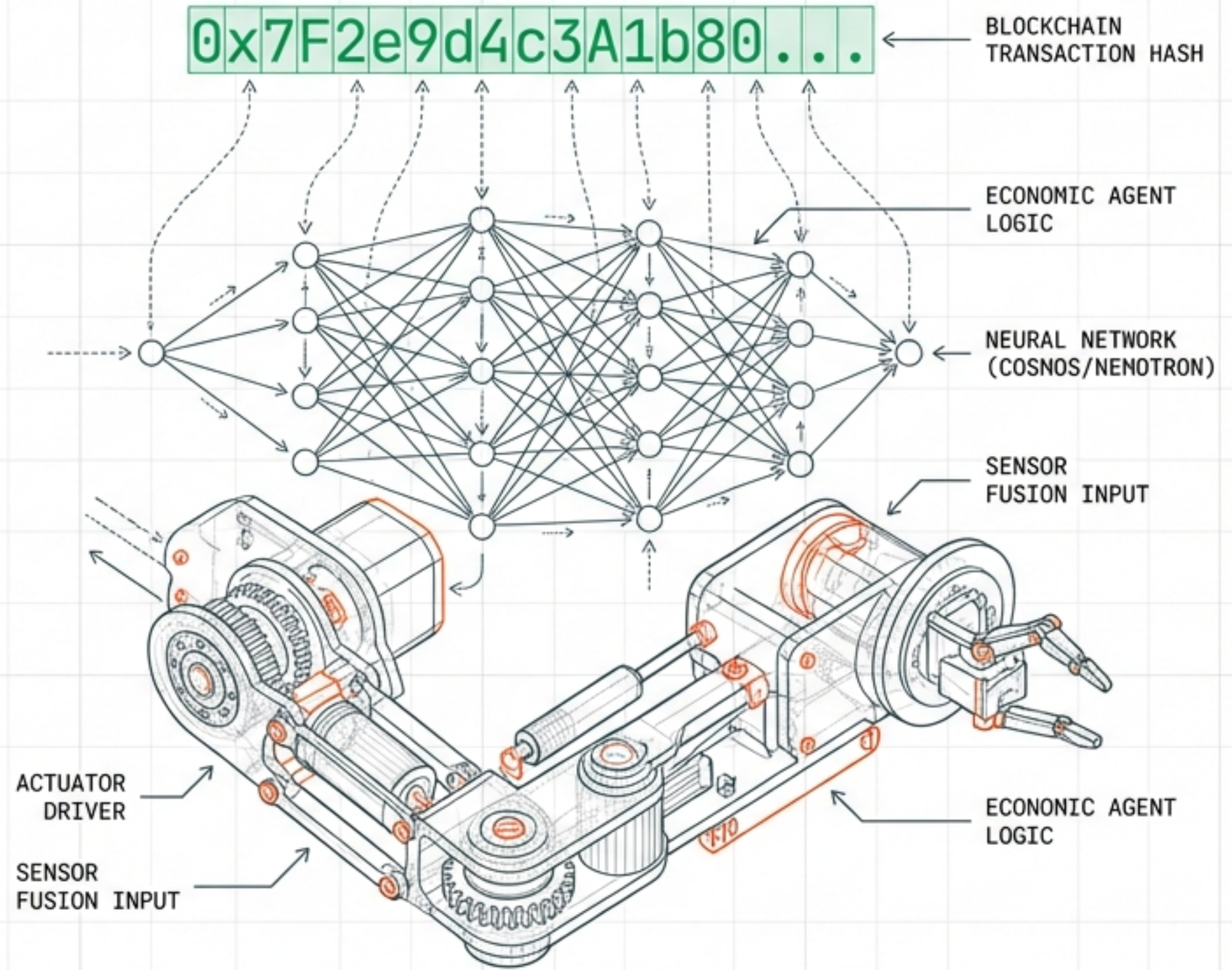


The Solana Robotics Kit

Building Autonomous,
Economic Agents on
NVIDIA Jetson Orin Nano.

Powered by:
NVIDIA Isaac
Cosmos & Nemotron
Solana Blockchain



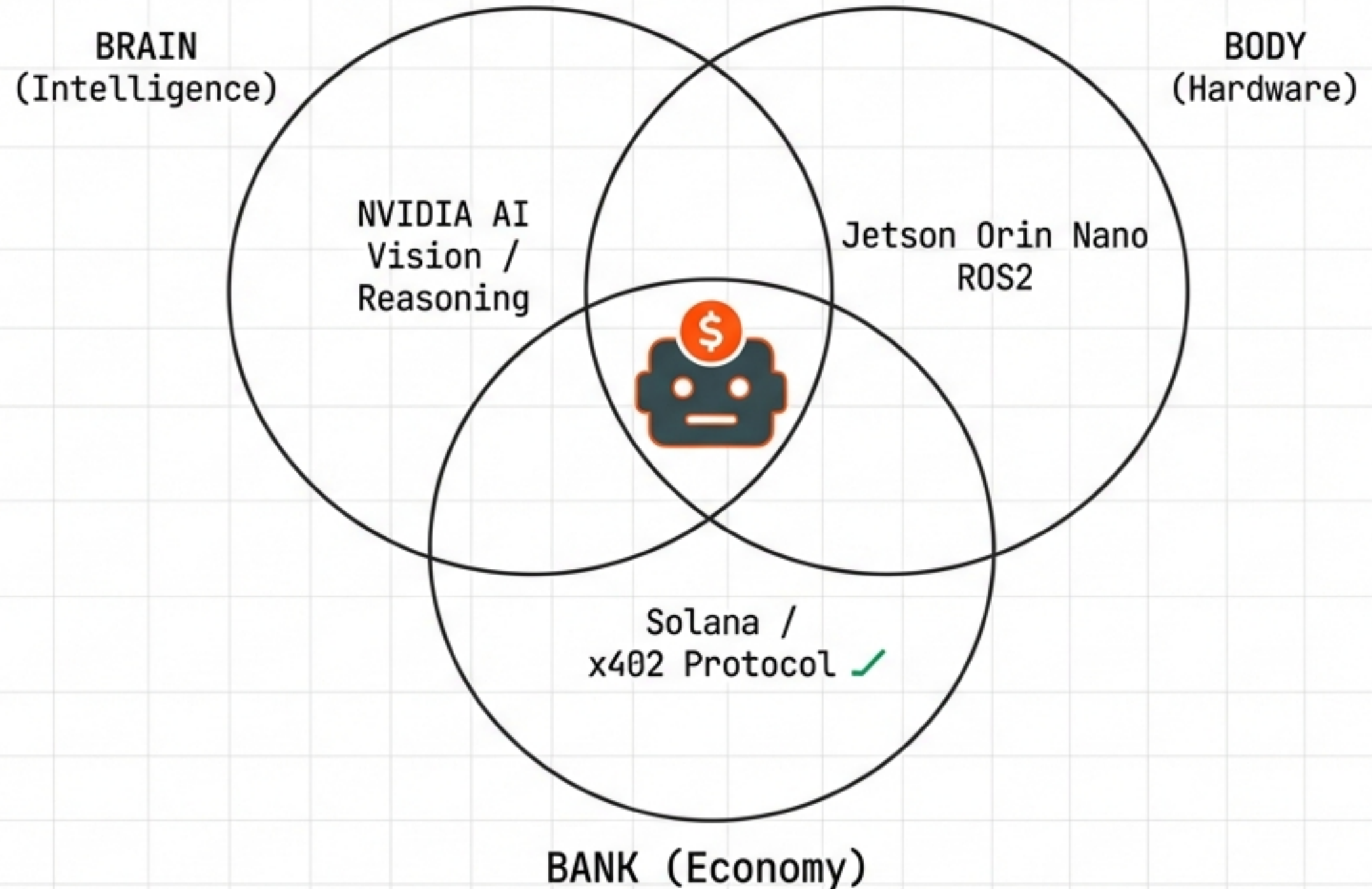
Version 1.0.0-alpha

THE EMBODIED ECONOMIC AGENT

We are shifting the paradigm from digital chatbots to physical entities.

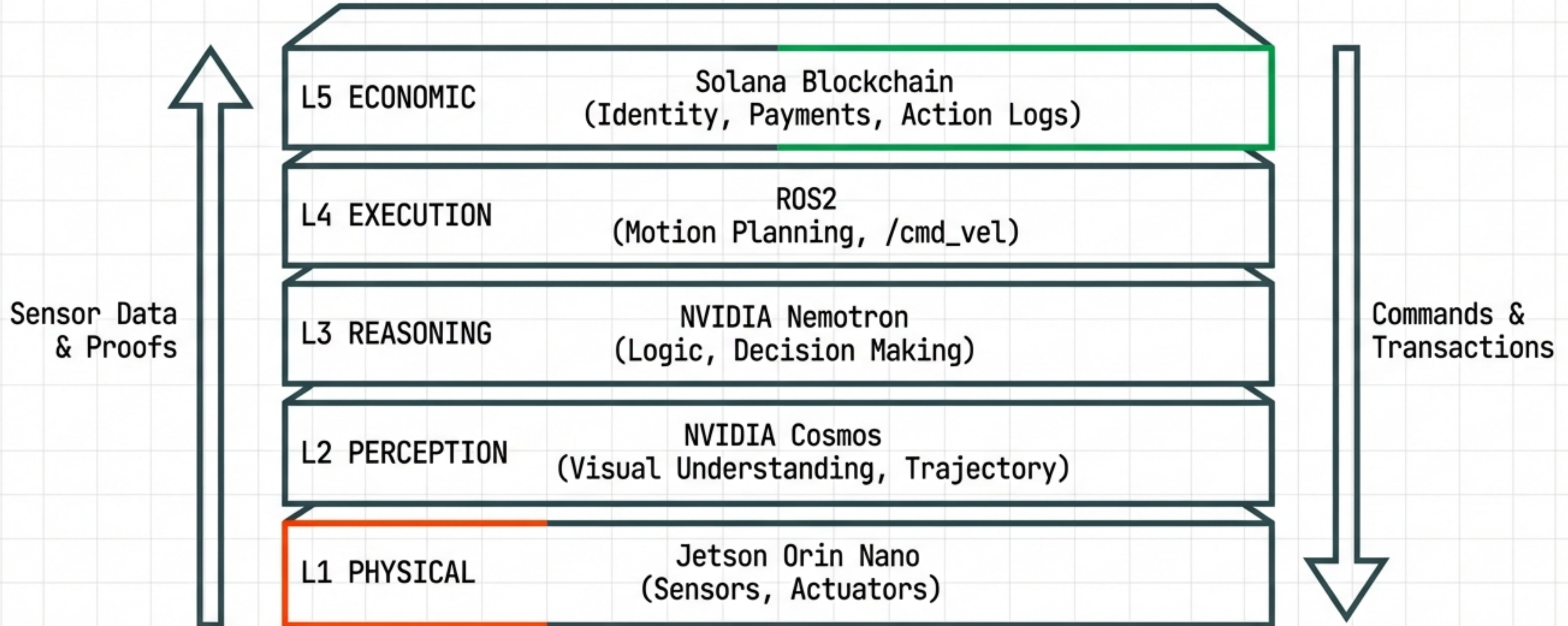
These agents do not just interact with the material world; they interact with the economic world simultaneously.

They perceive, act, and transact.



✓ Deliverable: v0.1 Starter Kit. Deploy an agent with a wallet, vision model, and movement controller out of the box.

Inter Tight: System Architecture: From Silicon to Settlement



Hardware Baseline: NVIDIA Jetson Orin Nano

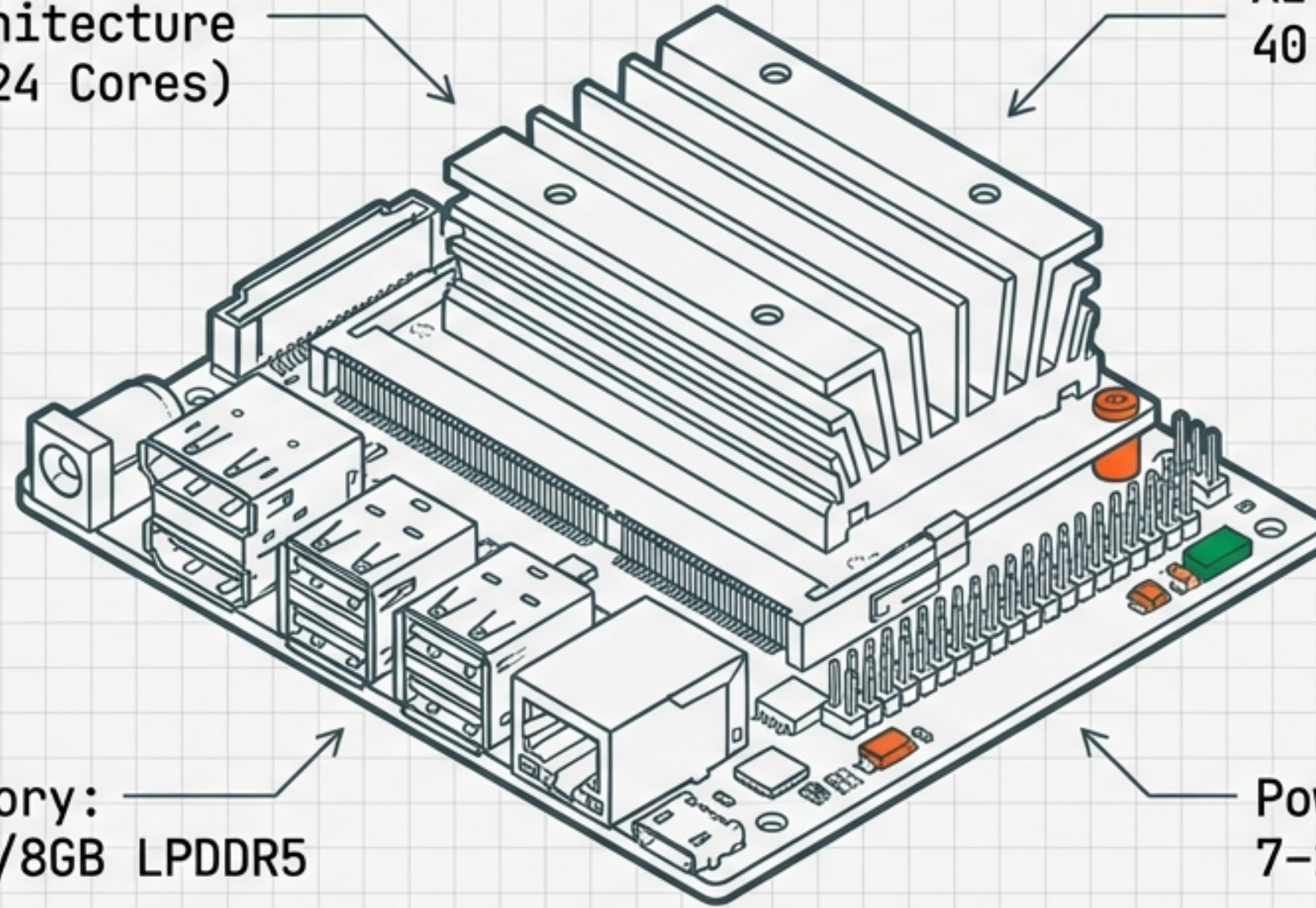
The Sweet Spot for Edge AI. Handles local ROS2 nodes, Isaac ROS acceleration, and safety-critical planning.

GPU: Ampere Architecture (1024 Cores)

AI Performance: 40 TOPS (INT8)

Memory: 4GB/8GB LPDDR5

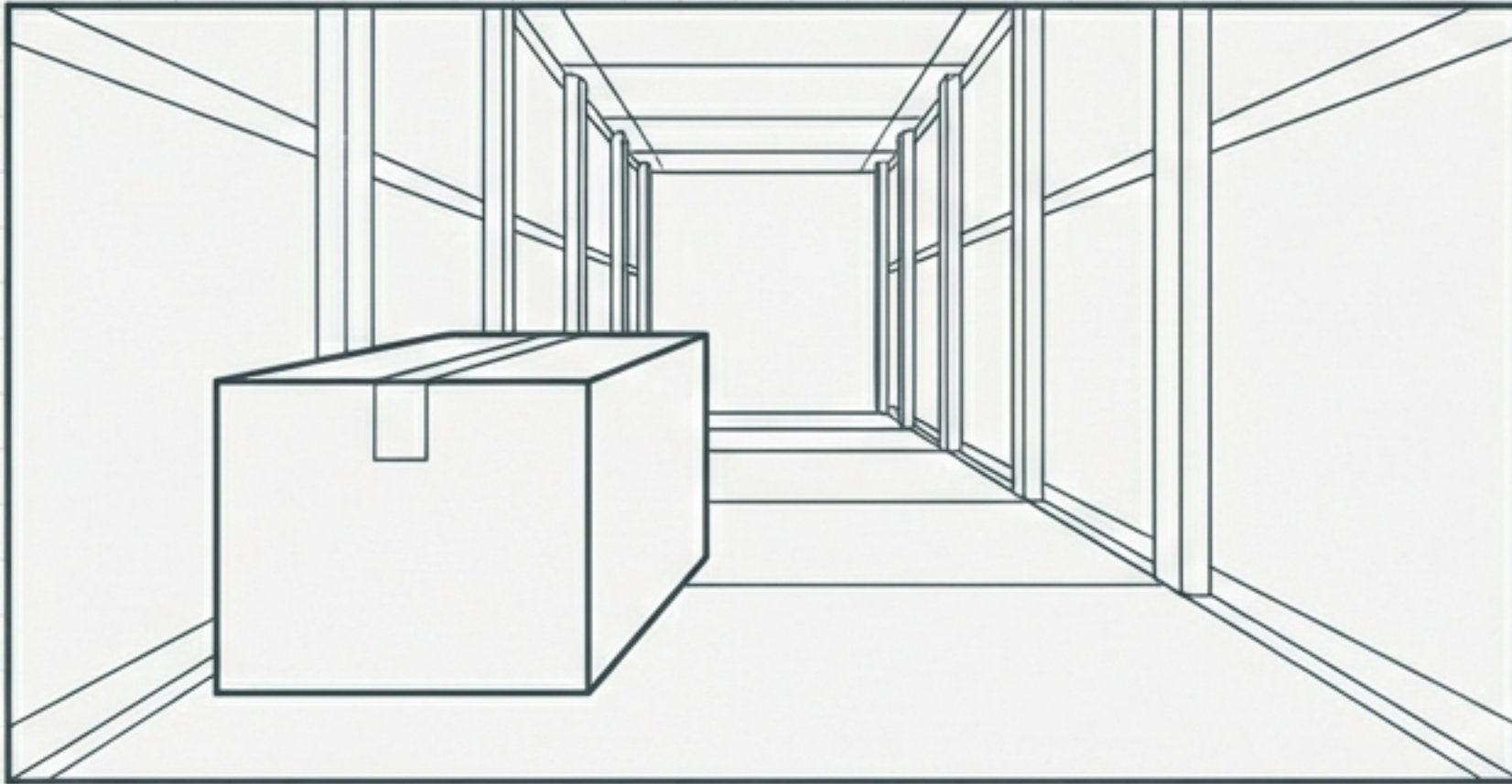
Power Efficiency: 7-15W



Strategy: Safety-critical loops run on-device. Heavy 30B+ inference offloaded to Cloud/NIM.

Perceive: Understanding, Not Just Seeing

RAW VISUAL INPUT

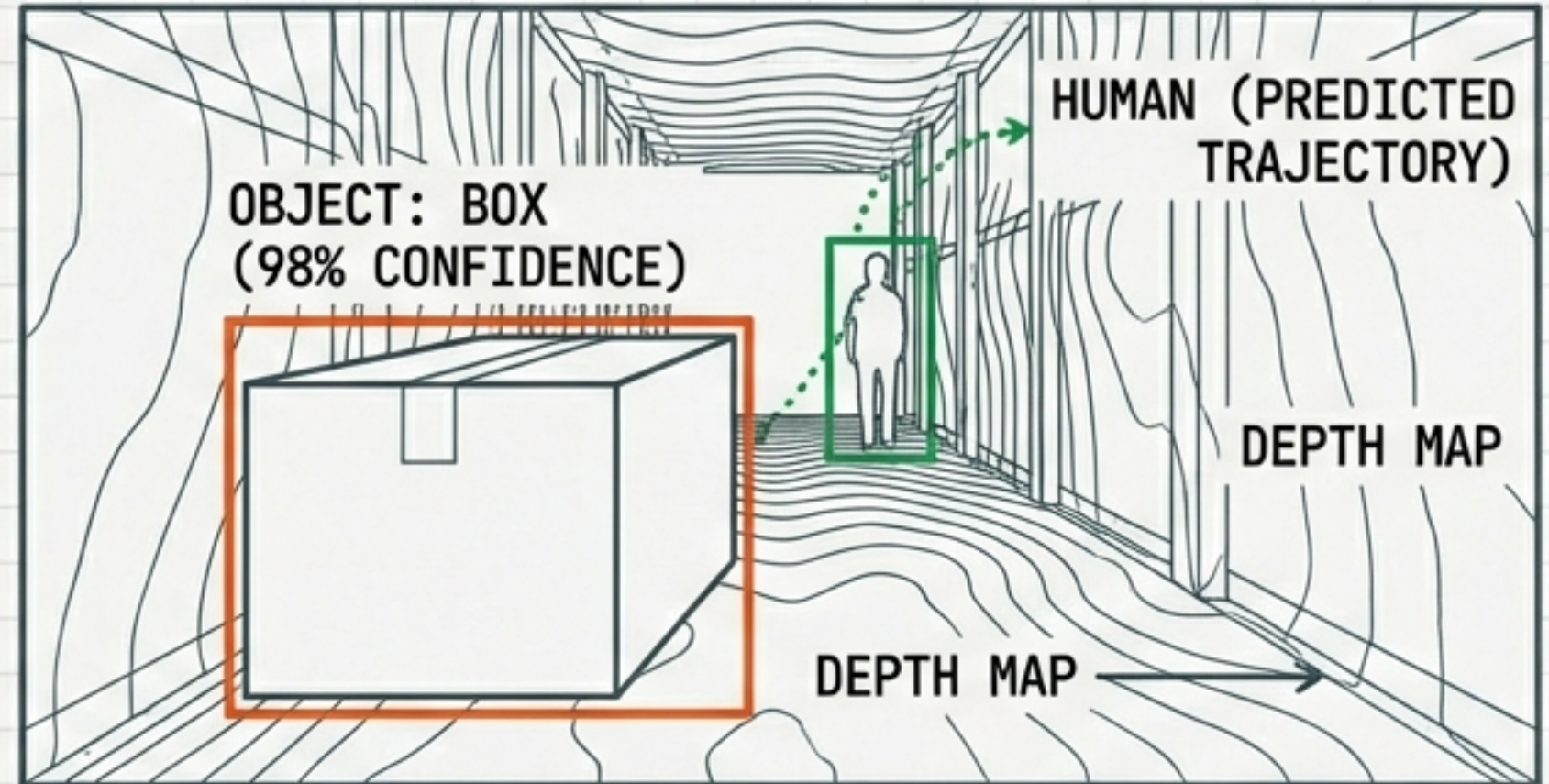


VISUAL REASONING

Model: cosmos-reason2-8b.

Function: Identifies objects, hazards, and spatial relationships. (e.g., 'Is this aisle blocked?')

COMPUTER VISION ANALYSIS



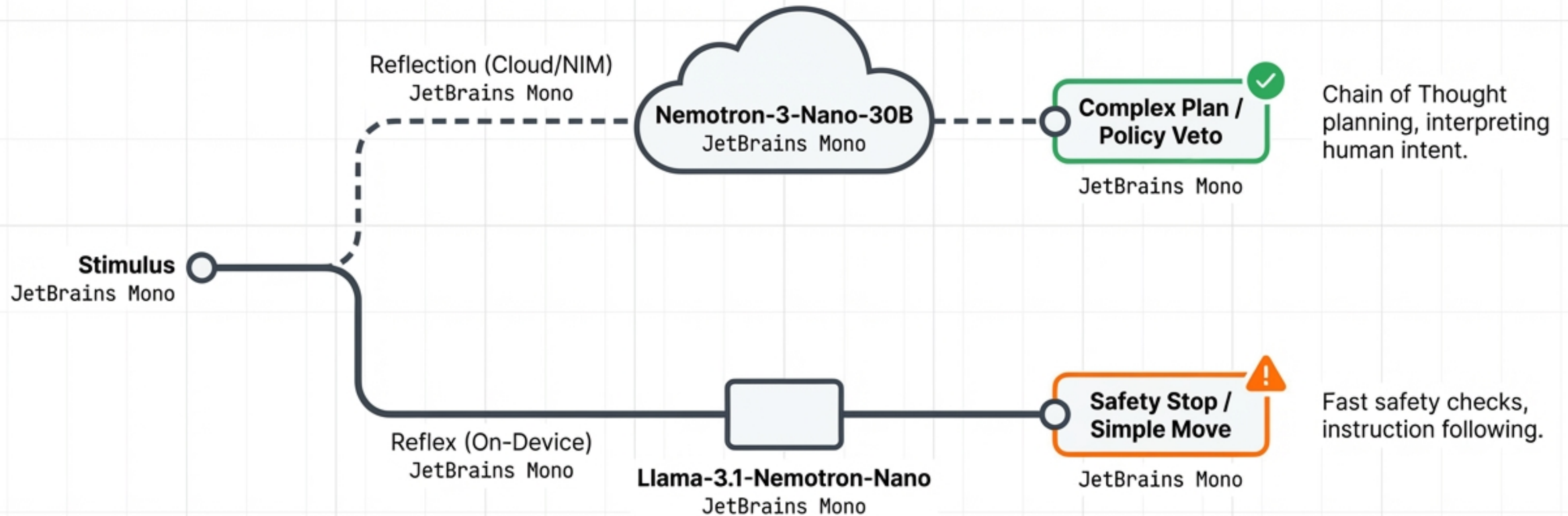
TRAJECTORY FORECASTING

Model: cosmos-predict1-5b.

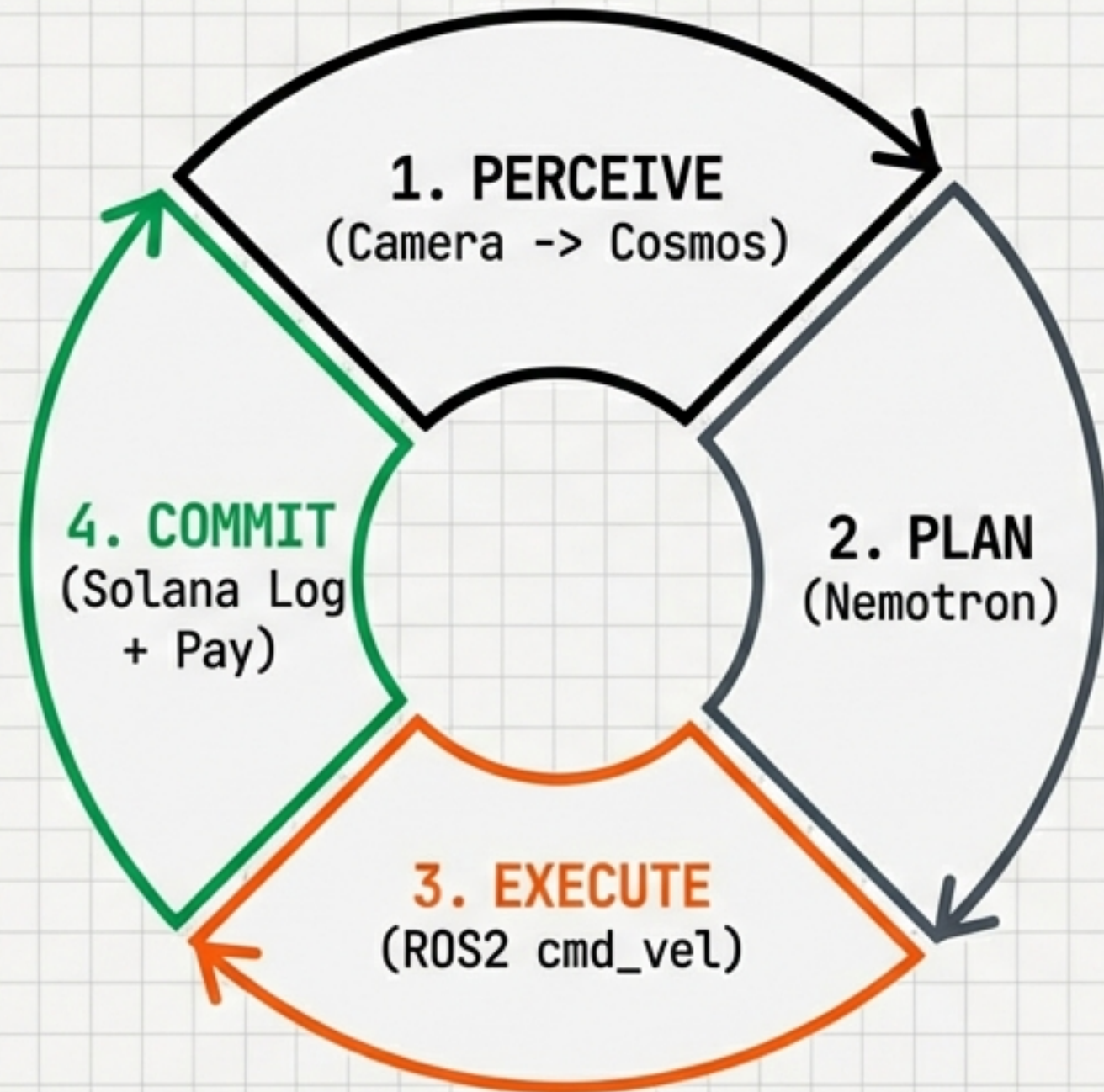
Function: Predicts future frames and human movement. (e.g., 'Where will this person be in 2 seconds?')

Integration via NVIDIA Integrate Endpoint.

Reason: The Hybrid Decision Engine



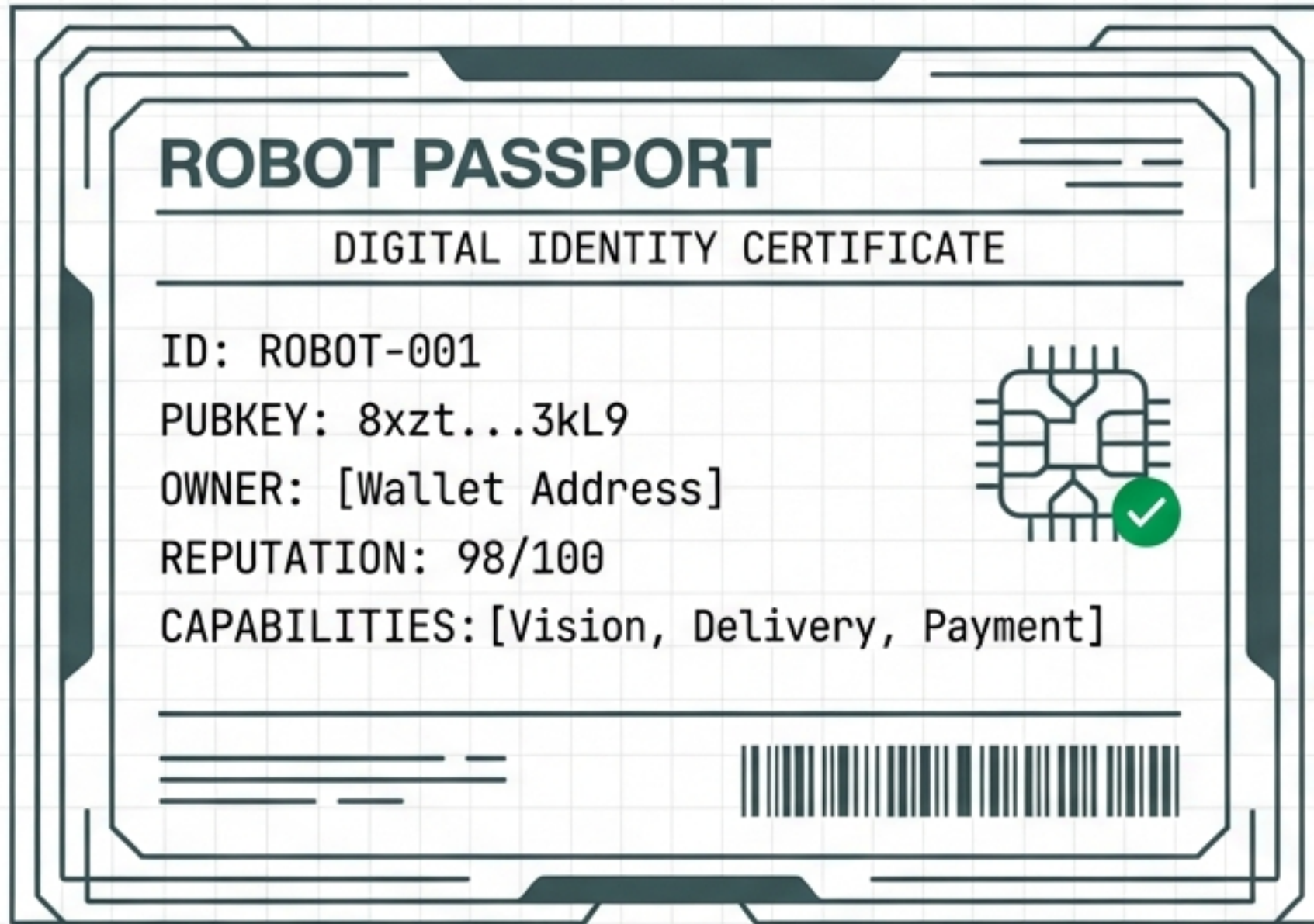
The Core Runtime Loop






PYTHON RUNTIME SNIPPET

```
while self.state.task_status == "running":  
    # The Infinite Loop of Autonomy  
    perception = await self.perceive(image_data)  
    decision = await self.reason(perception, task)  
    success = await self.execute(decision)  
  
    # The Economic Bridge  
    await self.log_action_onchain(decision)
```

The Economic Layer: Identity & Registry



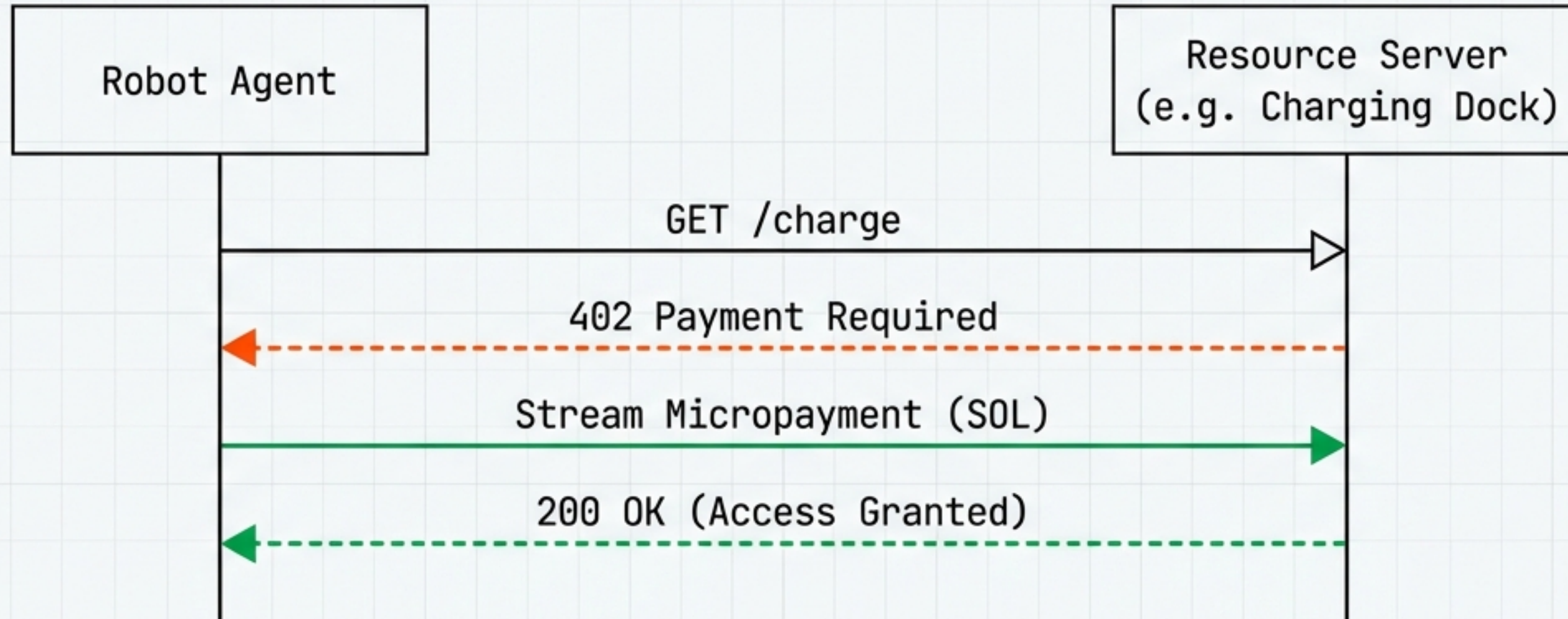
ON-CHAIN PROGRAMS

-  **Registry:** Stores metadata and ownership.
-  **Action Log:** Append-only event log for sensor proofs (Immutable).
-  **Marketplace:** Task bidding and escrow.

WHY SOLANA?

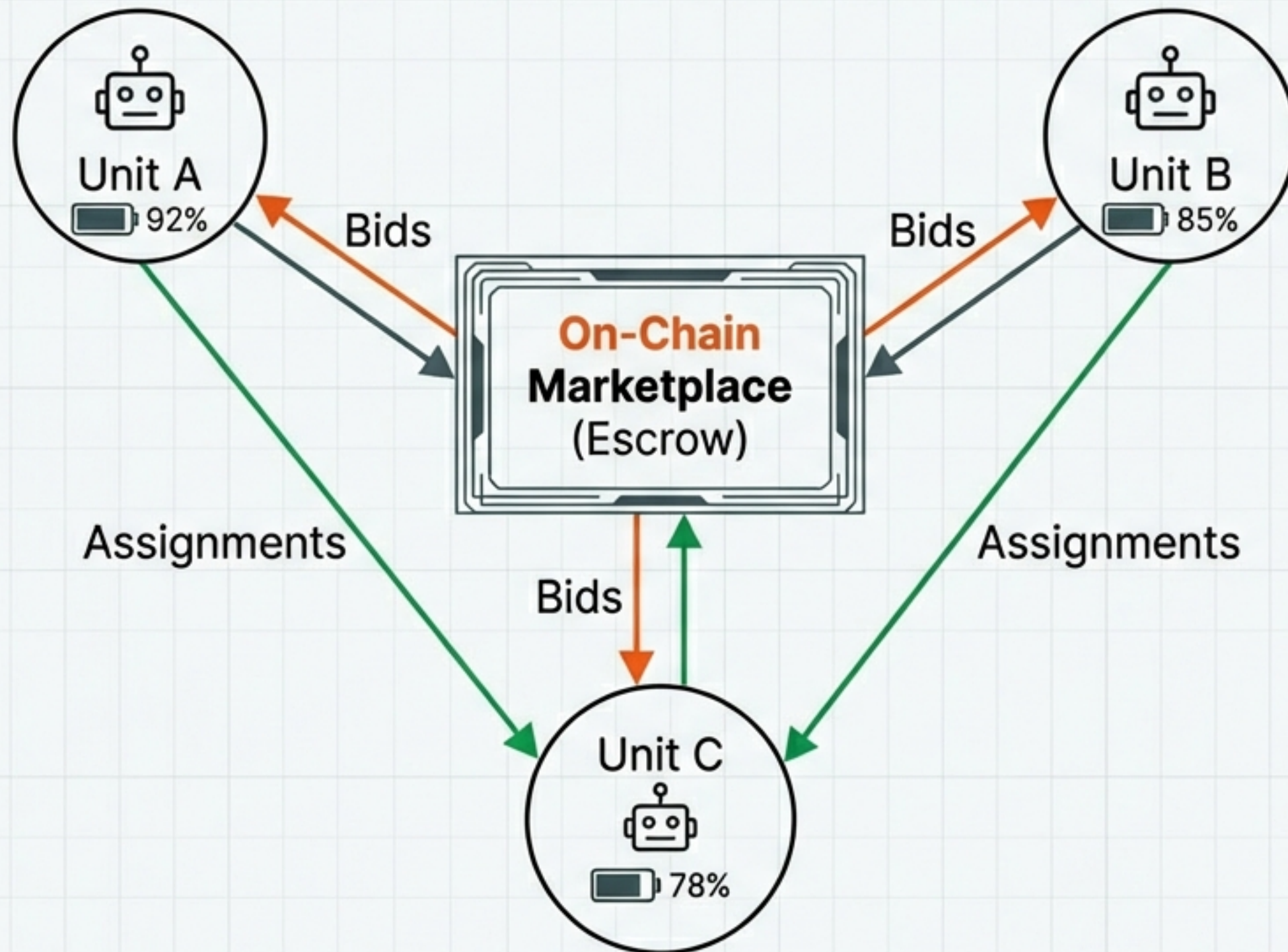
High throughput and low fees are essential for micro-transactions between machines.

The x402 Protocol: HTTP-Native Payments



Reviving the HTTP 402 status code. Robots handle payment challenges autonomously, enabling friction-free interaction with paid APIs and physical infrastructure.

Swarm Coordination & Task Allocation



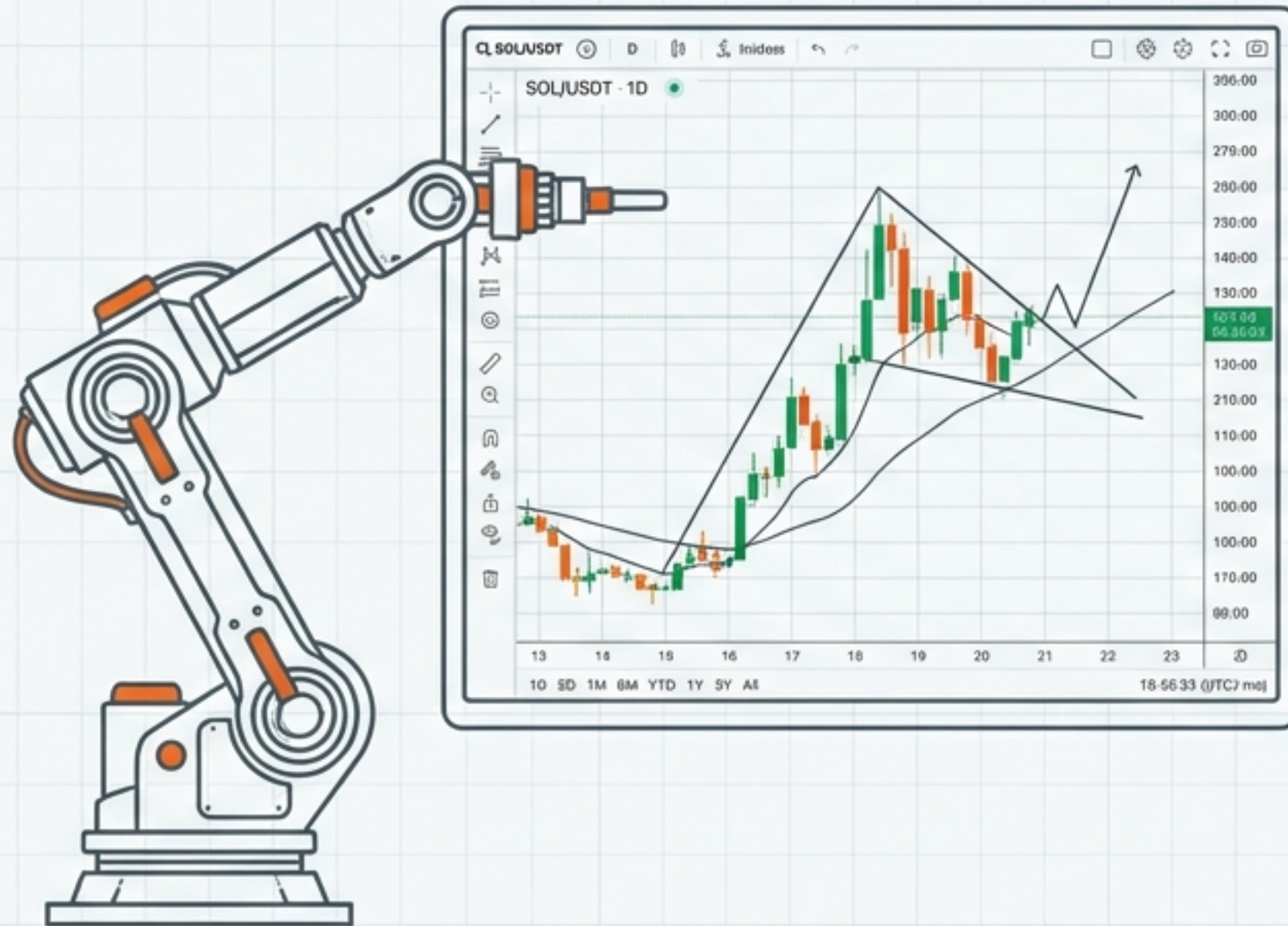
1. **Task Creation:** Coordinator posts task to Marketplace.
2. **Bidding:** Robots bid based on battery/proximity. Funds escrowed.
3. **Verification:** Sensor proofs logged upon completion.
4. **Settlement:** Payment distributed based on contribution.

Note:

Future State: Fully decentralized on-chain bidding logic.

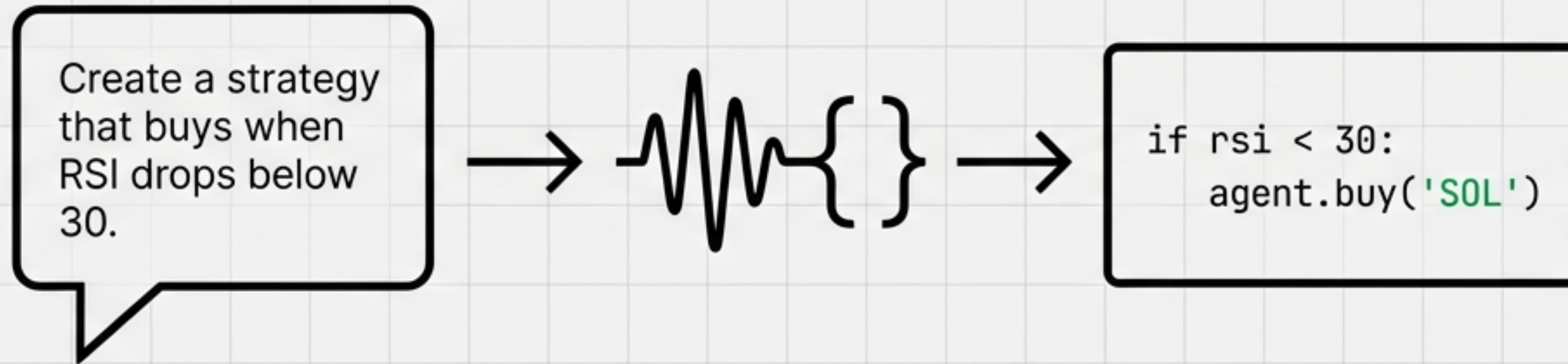
Flagship Application: Solana GROOT




An embodied AI trading robot. It points, explains rationale via voice, and physically/digitally executes the trade.



Vision	Voice	Action
✓ Cosmos Model reads chart patterns (Head & Shoulders, Wedges).	✓ Bi-directional comms via Parakeet/LiveKit. "Buy 10 SOL".	✓ Executes trades on Jupiter/Raydium DEX.

Interaction: Voice Control & Vibe Coding

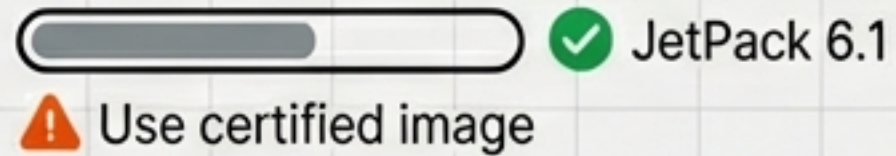
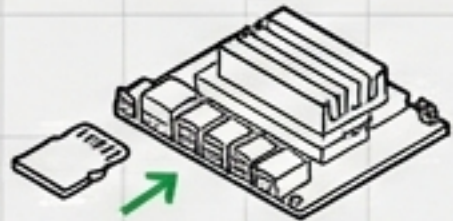


-  **Realtime Vibe Coding:** Natural language to executable Python.
-  **Voice Integration:** Powered by LiveKit & Parakeet.
-  **Safety:** Voice confirmation intercepts for high-value transactions.

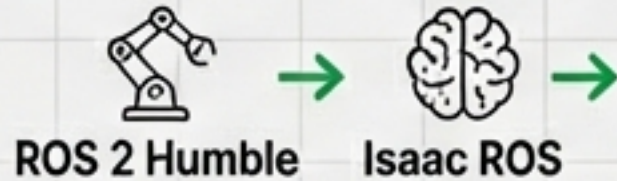
Developer Quickstart: Zero to Autonomy

The Happy Path

- 1 Flash **JetPack 6.1** on Orin Nano.

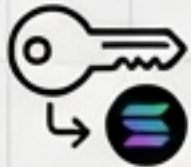


- 2 Install **ROS 2 Humble & Isaac ROS**.



```
$ sudo apt install ros-humble-desktop-full  
isaac_ros_common  
[SUCCESS] Installation complete.
```

- 3 Generate **Solana Keypair** (Devnet).

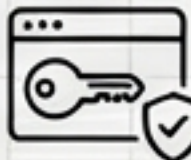


```
$ solana-keygen new --outfile ~/.config/solana/id.json  
wallet ..2853258d67 address 4d5...Z2b
```

Devnet

Keep private key secure!

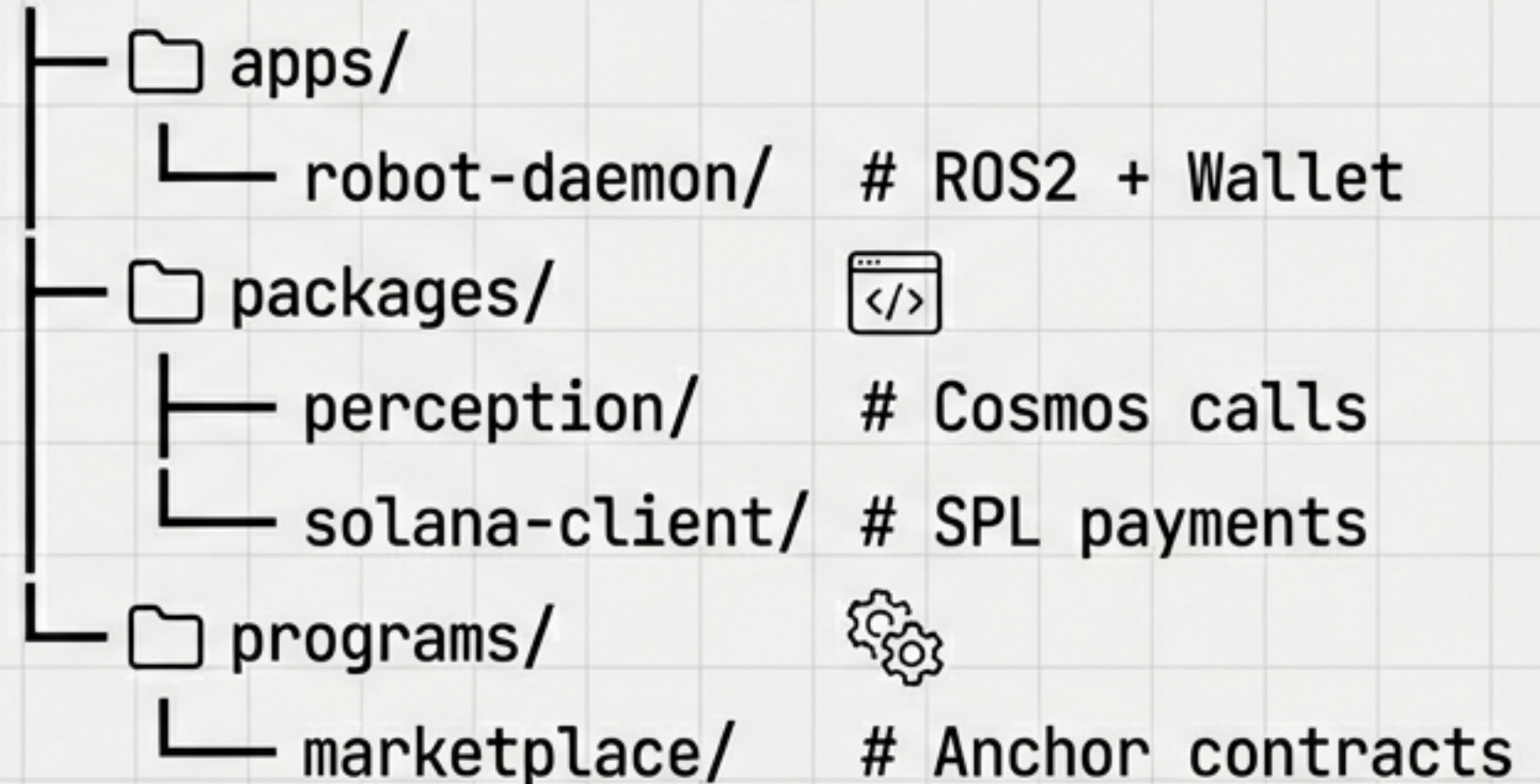
- 4 Set **NVIDIA_API_KEY**.



```
$ export NVIDIA_API_KEY=<your_key>  
[OK] Variable set.
```

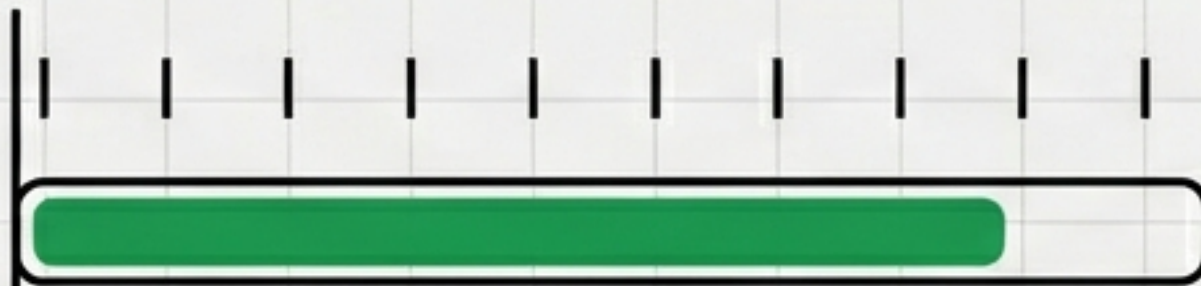
Repo Structure

solana-robotics-kit/



Roadmap: The Path to 2027

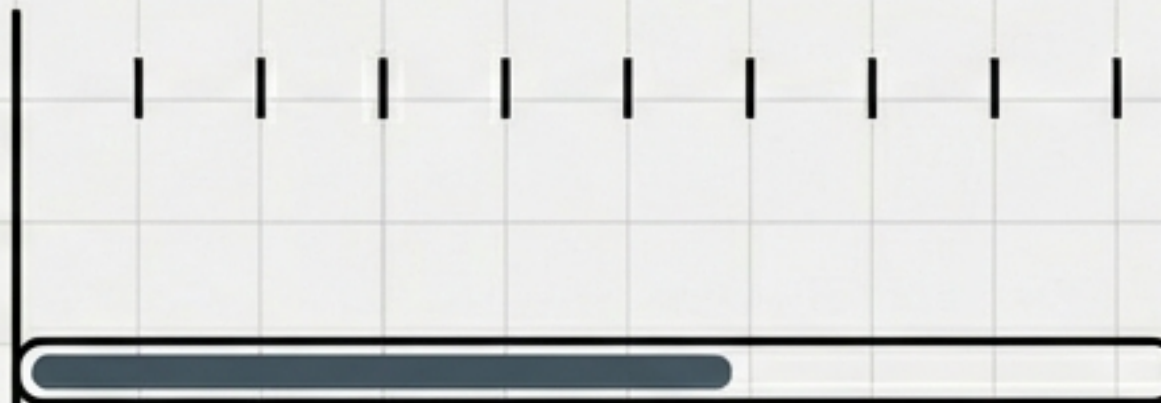
Q2 2026 (Alpha)



Current Status

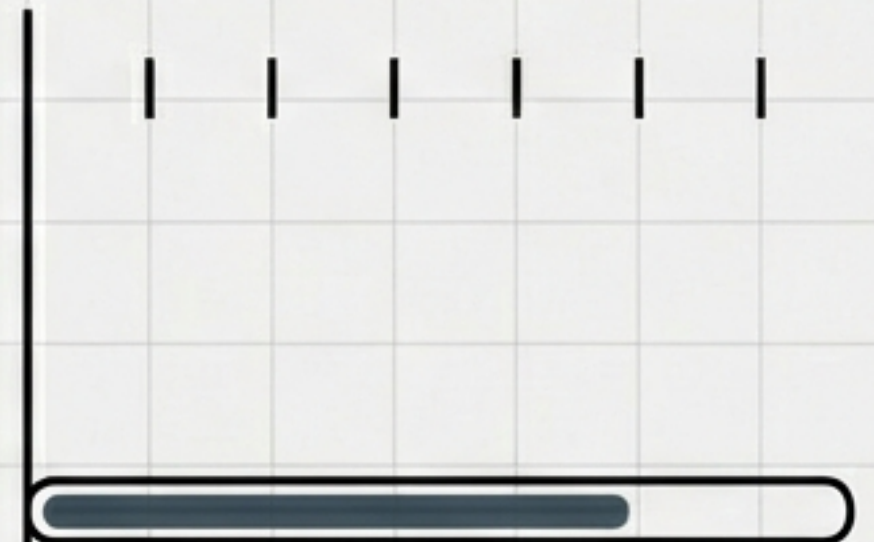
Basic navigation, ROS2 + Solana runtime, Cosmos vision.

Q3 2026 (Swarm)



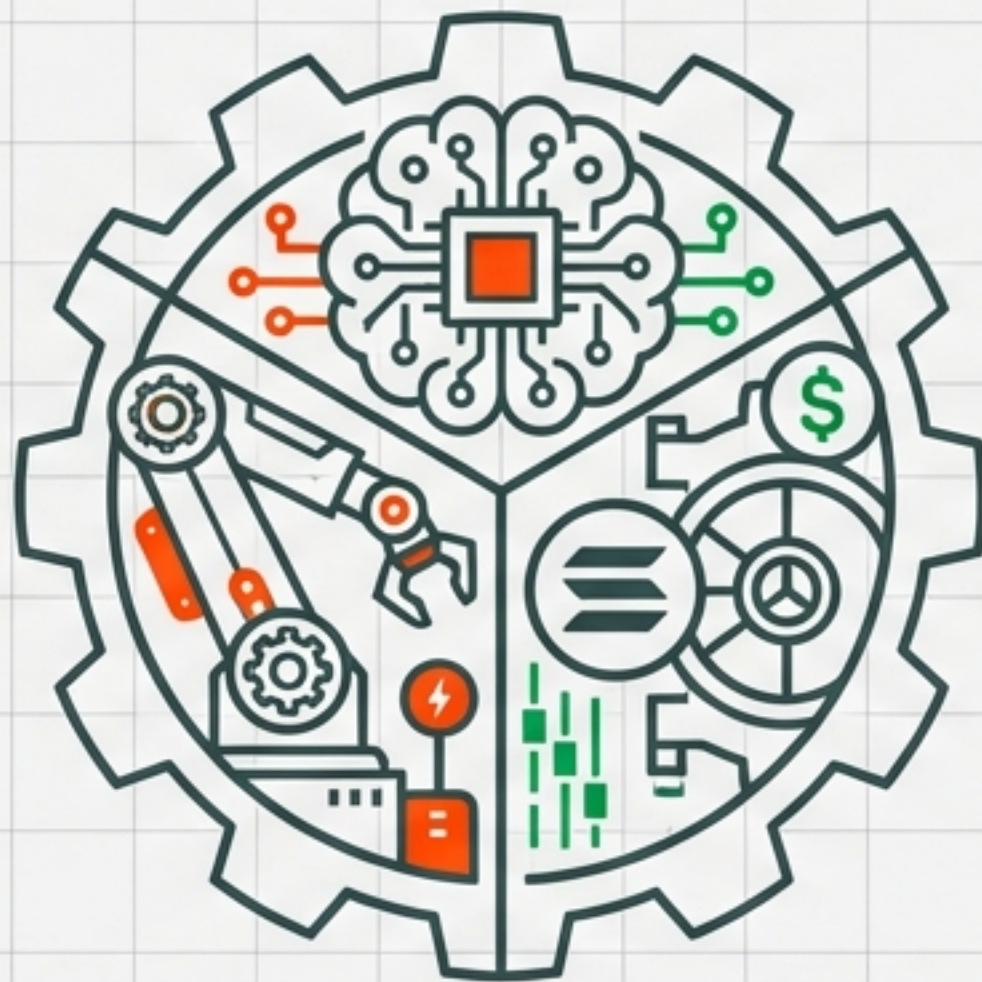
Warehouse automation demos, multi-robot coordination, on-chain marketplace.

2027 (Ecosystem)



Cross-chain robot networks, AI-driven manufacturing.

The Convergence is Here



Build the kit. Contribute to the repo.
Deploy the first generation of economic robots.

GitHub: 8bitsats/SolanaAutomation
Docs: NVIDIA Developer / Solana Docs
Community: Discord #solana-robotics