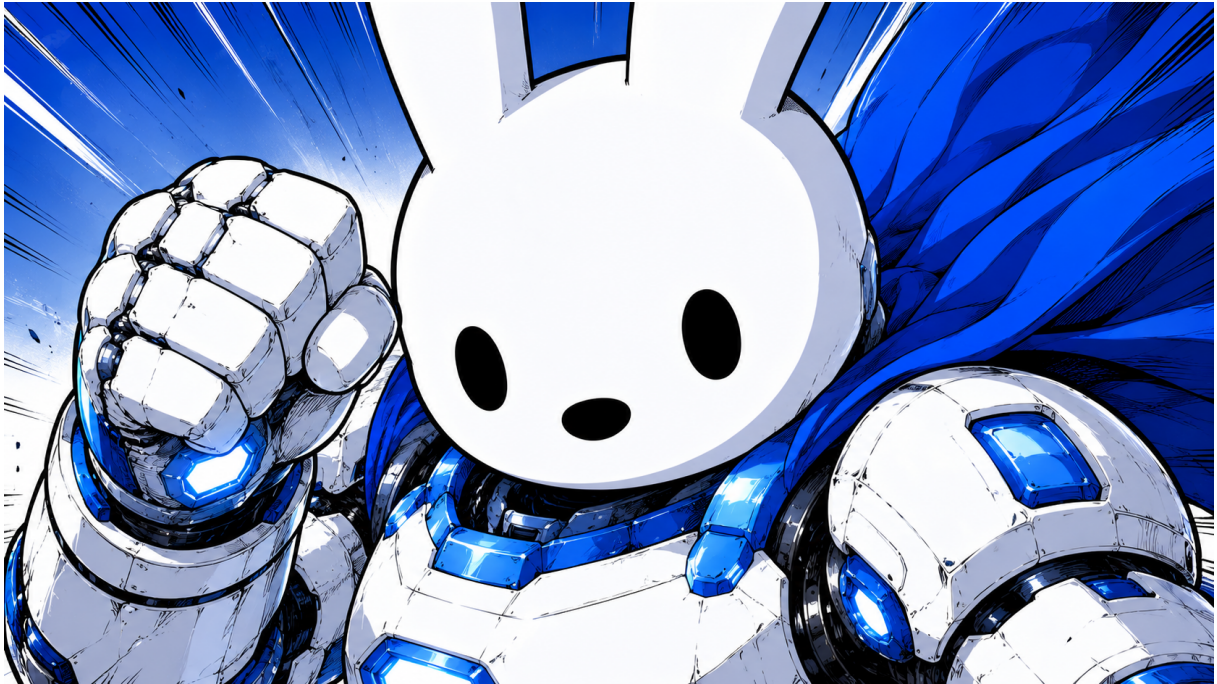

bunnyOS: The Small-Model Agent and Inference Platform

bunnyOS Foundation, Adam Ihsan, Joe Kawai

2026

Abstract

bunnyOS is a small-model agent and inference platform. It provides the infrastructure for running AI agents and applications powered by small language models, with compute supplied through bunnyCompute – a decentralised inference layer contributed by GPU owners worldwide. The platform is supported by a dual-token economy comprising OS and Carrot (CRT) on Base. This paper describes the architecture of the bunnyOS network and the design of its token economy.



1. What Are Small Models?

When we say “small models,” we are talking about highly optimized AI models typically ranging from 1 billion to 14 billion parameters, quantized to run efficiently on standard consumer hardware (e.g., Q4 format). They are fast, lean, and deployable without specialized infrastructure.

For most of their history, small models were limited to narrow, well-defined tasks: classification, summarization, simple Q&A. The more demanding work – reasoning, planning, acting autonomously across complex workflows – required much larger models running on specialized hardware.

That has changed.

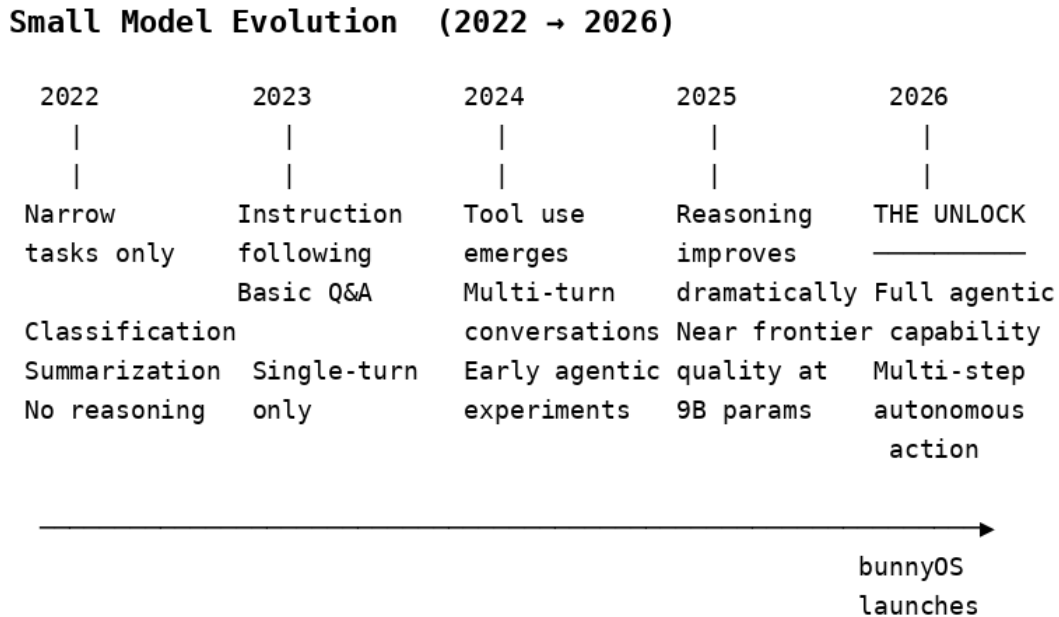


Figure 1: Small model capability evolution from 2022 to the 2026 agentic unlock.

In 2026, state-of-the-art small models have crossed a threshold. Models like Gemma 4 and Qwen 3.5 at 4–9 billion parameters are now capable of performing genuine agentic tasks: multi-step reasoning, tool use, autonomous decision-making, and real-world action. The capability gap between small and large models is closing rapidly, while the cost and accessibility gap remains enormous.

Small models are not a compromise. They are a different and increasingly powerful category of AI infrastructure – one that can run close to the user, on hardware that individuals and small businesses already own.

2. The Opportunity

Since the year 2000, approximately 2.3 billion discrete graphics cards have been sold worldwide. That is 2.3 billion units of compute sitting in homes, offices, and gaming rigs around the world, most of it idle for the majority of every day.

People have been trying to unlock this latent capacity for decades. Distributed computing projects, crypto mining, and decentralised rendering networks all attempted to put idle GPUs to work. But the use cases were always niche. None of these projects created something that billions of people actually needed to use continuously.

The missing ingredient was not compute. It was **demand**.

In 2026, small-model AI agents are the workload that idle consumer GPUs have been waiting for. Small models are perfectly matched to consumer GPU hardware: efficient,

fast, and needed continuously by anyone running an agent.

The opportunity is to **unlock capability, capacity, and demand simultaneously**.

bunnyOS unlocks **capability** by building the platform, agents, and apps that make small models genuinely useful for everyday tasks. It unlocks **capacity** by incentivising GPU owners worldwide to contribute their hardware to the network. And it unlocks **demand** by creating a network where running agents and building apps creates real, continuous usage – turning idle consumer hardware into a yield-generating asset for contributors and stakers alike.

Whoever builds the default infrastructure for small-model agents will define the next era of AI. That is what bunnyOS is building.

3. What Is bunnyOS

bunnyOS is a small-model agent and inference platform. Our mission is to push the frontier of what is possible with small models – building the demand, providing the supply, and creating the economic layer that makes it all sustainable.

The bunnyOS network consists of four core components:

3.1 bunnyOS: Agents and Apps

The home for small-model agents and applications. Users access a growing network of agents and apps – tools, services, and games – all running natively on small-model infrastructure. bunnyOS is where demand lives.

3.2 bunnyAgent: Small-Model Optimised Agents

bunnyAgent is a general agent optimised for small models. It is capable of reasoning, planning, and taking action across a wide range of tasks, running entirely on small-model inference. bunnyAgent is also open-source, giving developers the harness they need to build their own agents on top of the same foundation: memory management, tool routing, context compression, and native integration with the bunnyOS network.

3.3 bunnyApps: Apps That Use Small Models

Apps that use small models – tools, services, and games. Developers build bunnyApps to deliver experiences and utilities that run natively on small-model inference. Whether shipping a productivity tool, a data service, or an AI-powered game, bunnyApps are the primary vehicle for delivering value to users on the platform.

3.4 bunnyCompute: Decentralised Small-Model Inference

The decentralised small-model inference provider. bunnyCompute aggregates GPU capacity from contributors worldwide and routes inference requests to the most efficient available hardware. It is purpose-built for small models – optimised for the specific quantization formats, latency profiles, and throughput characteristics that small-model workloads demand. bunnyCompute is where supply lives.

bunnyOS Platform Architecture

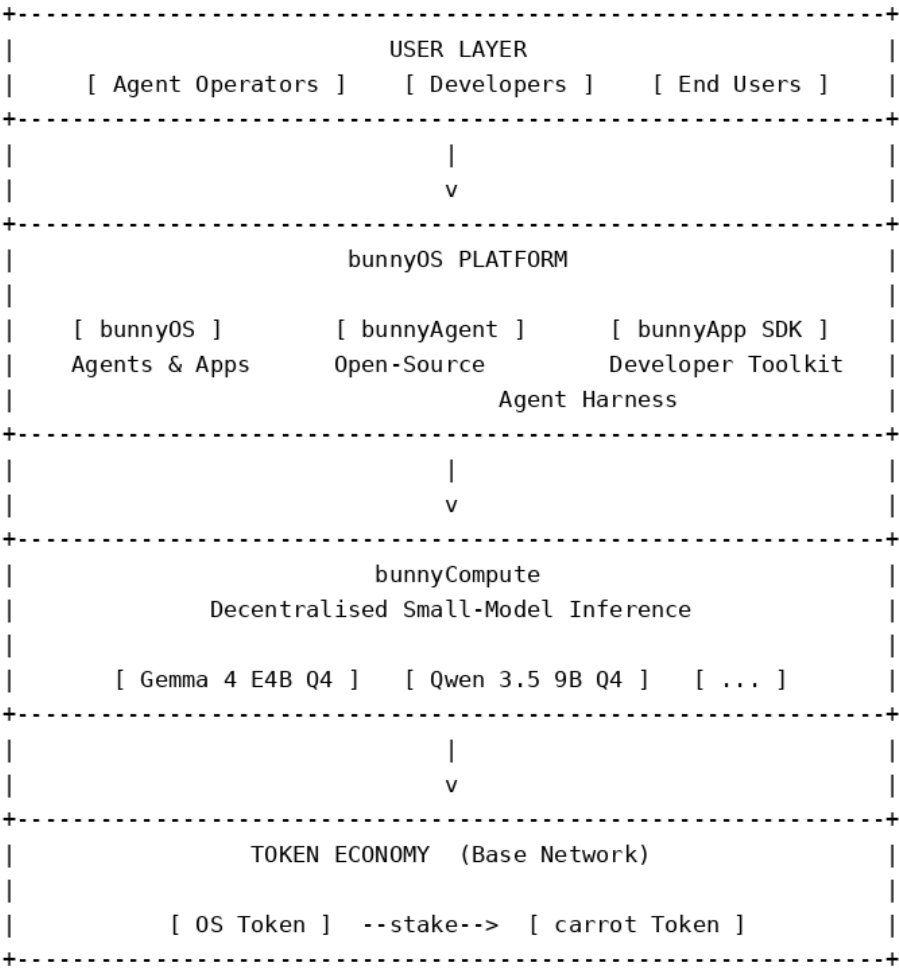


Figure 2: The bunnyOS platform architecture.

Together, these four pillars form a closed loop: bunnyOS builds the demand, bunnyCompute provides the supply, bunnyAgent gives developers and users the agents they need, and bunnyApps give developers the tools to build products on top of the network.

4. The Token Economy

The bunnyOS network is powered by two tokens on Base: **OS** and **Carrot (CRT)**. Each has a distinct role, and together they form the economic backbone of the network.

Token Flow: OS → carrot → Services

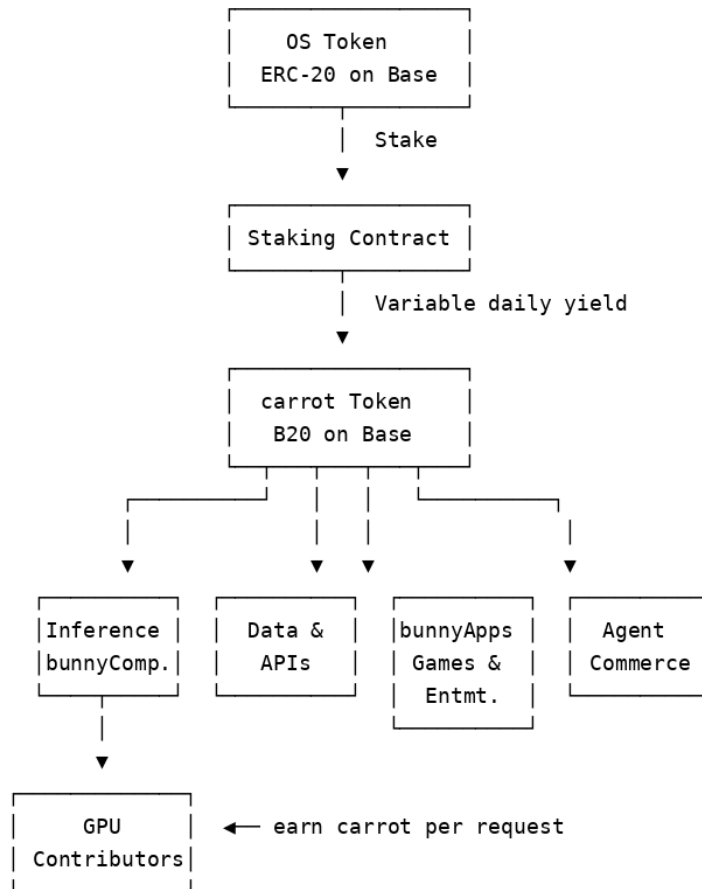


Figure 3: Token flow: OS staked to earn CRT, CRT used across network services.

4.1 OS: The Service Capacity Token

OS is the service capacity token of bunnyOS. It is an ERC-20 token on Base.

Think of the bunnyOS network as a factory. **OS** is your stake in the factory’s productive capacity. **CRT** is what the factory produces. The factory’s product is inference and services. Stake **OS**, and the factory produces **CRT** for you – proportional to your stake, for as long as you hold it.

Staking **OS** entitles you to a proportional share of the bunnyOS network’s daily **CRT** output. The yield is variable, reflecting the health and activity of the network: as usage grows, staking becomes more productive. As the network expands, so does the value of holding **OS**.

OS is not equity, not governance, and not a loan. It is a stake in the network’s obligation to produce. The more the network is used, the more **CRT** flows to **OS** stakers.

4.2 Carrot (CRT): The Unit of Service

Carrot (CRT) is the unit of service on the bunnyOS network. It is a B20 asset on Base.

CRT is used to access inference, data, and services across the network. Every agent action, every API call, every interaction with a bunnyApp runs on **CRT**. Think of it as owned capacity – when you hold **CRT**, you hold the ability to run your agents on demand, without a billing cycle.

Some people will use **CRT** the moment they earn it. Others will hold it, save it, or pass it on. **CRT** is transferable, and its value is determined entirely by the network – by how much people want to use it and how much is available. bunnyOS does not set a price for **CRT**. The network does.

Token	Role	How You Get It	What You Do With It
OS	Service capacity token	Buy or trade on Base	Stake to earn CRT daily; proportional to your share of staked OS
CRT	Unit of service	Stake OS, contribute compute, or acquire on Base	Use for inference, data, and services across the network

4.3 The Economic Flywheel

The bunnyOS economy is grounded in real revenue. bunnyCompute sells decentralised inference to developers and users via API access and subscriptions. That revenue powers **CRT** emission to GPU contributors and **OS** stakers. **CRT** is then used by agents and users to access services, creating continuous demand that gives the token real utility.

The loop is as follows:

Real compute revenue → CRT emission → staking yield for OS holders → CRT used on services → demand for more compute → more revenue.

Every part of the loop is anchored to real usage. **CRT** is not minted arbitrarily; it is earned by contributing to and participating in a network that people are actively paying to use.

The bunnyOS Economic Flywheel

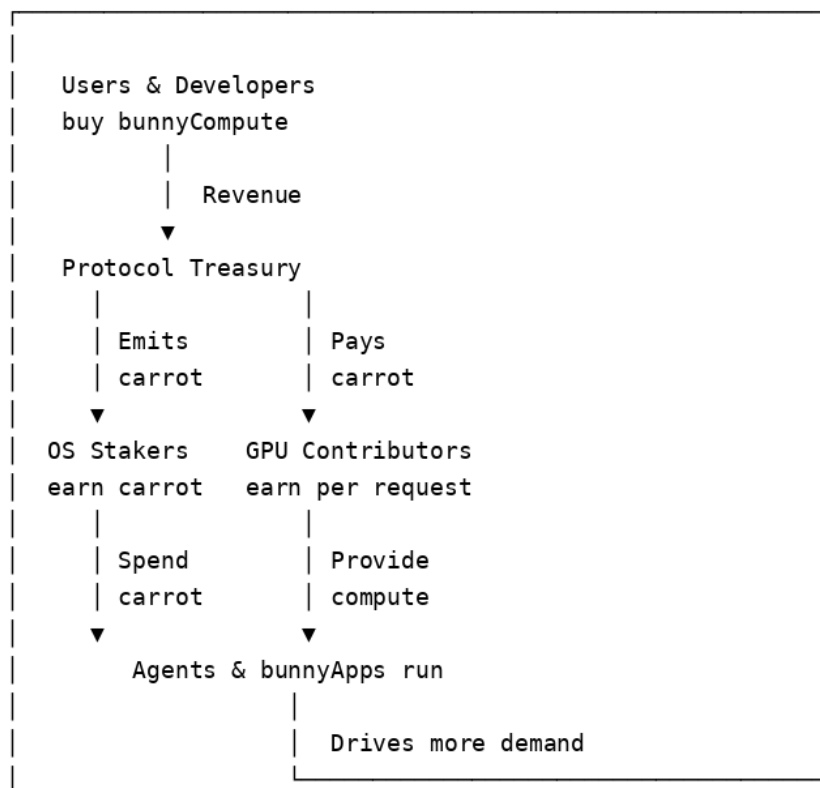


Figure 4: The bunnyOS economic flywheel.

5. Conclusion

We are at the beginning of the agentic era. In a few years, every person, every developer, and every business will have agents working for them continuously.

Mission and Vision

bunnyOS exists to unlock the powers of small models. Our vision is to build the demand and supply the inference that enables small-model intelligence to reach its full potential – for humans, agents, and robots. We are not building for a single use case or a single generation of AI. We are building the foundational layer that will serve every autonomous system that runs on small-model inference, from the agent on your phone today to the robot in your home tomorrow.