

Open 816B Weights, Cheaper Flash Models and the KV-Cache Squeeze: Meta Muse Spark Open, MiniMax M3.1 Flash and DeepSeek V4.1-Flash

Meta's Muse Spark Open, an 816-billion-parameter open-weight model run on clustered Mac Studios, MiniMax's M3.1 Flash preview aimed at agent and coding work, and DeepSeek's V4.1-Flash technical report on compressing the KV cache.

Themes: Models & Frontier · Infrastructure & Economics

Note: primary pages could not be fetched this run, so details come from search summaries of secondary outlets. Vendor-reported benchmarks are flagged. This corrects Day 205, which left out the Meta 816B claim as unconfirmed: multiple outlets now report it.

1) Meta Muse Spark Open: A Frontier-Sized Open-Weight Model

Meta released Muse Spark Open on October 15. Reports describe 816 billion total parameters with 42 billion active per token, a 1 million-token context window and native multimodality. Meta calls it the most capable US open-weight model and says it worked with Apple so it runs locally on two 512 GB or four 256 GB M5 Ultra Mac Studios using Apple's MLX framework. Weights are on Hugging Face and cloud catalogues (SageMaker, Google Model Garden, Microsoft Foundry), under the Muse Model License, which has limited commercial terms. Meta's own chart reports leads on MCP-Atlas (89.2) and JobBench (61.8) but trails Kimi K3 and GLM 5.3 on Terminal-Bench 2.1 (76.4 vs 85.0 and 83.9); these are vendor-reported and not independently confirmed.

The concept, explained simply: A mixture-of-experts (MoE) model has many specialist sub-networks but routes each token through only a few. Here 816B parameters must sit in memory, but only 42B do work per token, so it is far faster than a dense model of the same size. That is why it needs about 1 TB of memory yet can still run on desktop machines.

Why it matters: Open weights at this scale let companies run a near-frontier model on their own hardware, keeping data private. The catch is memory, not compute: the cost has moved from per-token API fees to owning roughly a terabyte of unified memory, and the licence is not fully open source.

2) MiniMax M3.1 Flash: Faster, Cheaper Agent Models

MiniMax's M3.1 Flash is reported as an update aimed at agentic tasks, coding, computer use and cybersecurity, with claims of roughly 3x faster inference and 70% lower usage cost than its predecessor. Details are murky: it appeared as a preview in MiniMax Code, one source lists no pay-as-you-go API price (access via subscription tiers), and no independent benchmarks or model card were found. Treat the speed and cost claims as vendor-reported.

The concept, explained simply: A 'Flash' model is a smaller, faster sibling of a flagship model, tuned for high-volume work. Agents call a model dozens of times per task, so speed and cost per call matter more than peak intelligence.

Why it matters: Flash-class models are where most real agent spending will go. Chinese labs keep shipping them quickly, which pressures prices everywhere. Also note the pricing model: subscription-only access limits how builders can meter and budget.

3) DeepSeek V4.1-Flash and the KV-Cache Squeeze

DeepSeek's V4.1-Flash family (first released September 10) comes with a technical report titled 'Pushing the Limits of KV Cache Compression'. DeepSeek reports 74.2% on Deep-SWE v1.1 (up from 54.4% for V4-Flash) and 90.6% on Terminal-Bench 2.1; these are the company's own figures. The October 17 update, V4.1-Flash-1017, was covered on Day 205. Parameter counts are reported inconsistently, so they are left out here.

The concept, explained simply: While generating text, a model stores a key and a value vector for every earlier token so it need not recompute them: the KV cache. For long contexts it can grow larger than the model itself and becomes the main memory and cost bottleneck. Compressing it means storing fewer or smaller vectors per token while keeping quality.

Why it matters: Long-context agents (reading codebases, running for hours) are limited by KV-cache memory. Compression is how a 1M-token window becomes affordable, and it ties today's three stories together: Muse Spark Open's 1M window and 1 TB footprint, and Flash models' price cuts, all depend on memory efficiency.

VIRAL / OPEN-SOURCE SPOTLIGHT

Muse Spark Open on a Mac Studio cluster (Hugging Face + MLX). The trend of the week is running a frontier-class open model at home: two linked Mac Studios give about 1 TB of unified memory and run an 816B model locally, with a cookbook for setup. Why it is taking off: it combines a big-name release, consumer hardware and a free download, so developers share their own clusters and speed tests. Caveats: it needs a very costly setup (one outlet notes the 512 GB Mac Studio is not yet sold by Apple), reported speed is slow (the cookbook estimates hours for a long task and says four Macs serve fastest), and the licence limits commercial use.

MARKET SIGNAL

The contest is moving from bigger models to cheaper and more private ones. Open frontier weights, Flash-class agent models and cache compression all push the cost of a task down, while hardware makers (Apple here) and cloud catalogues compete to be where open models run. Expect buyers to ask what a task costs end to end, not what a token costs.

PRACTICAL TAKEAWAYS

1) Size memory before you size models.

For open MoE models, total parameters set the memory you need and active parameters set the speed, so check both before buying hardware.

2) Read the licence, not just 'open'.

Muse Model License limits commercial use, so have legal review it before building a product on the weights.

3) Test vendor speed and cost claims on your workload.

Claims like 3x faster and 70% cheaper are unaudited, so run your own tasks and compare cost per completed task.

4) Watch context length costs.

A 1M-token window is only useful if the KV cache fits, so prefer models and servers with cache compression for long agent runs.

SELF-TEST

Q1. What is a mixture-of-experts model?

A1. A model with many specialist sub-networks where only a few process each token, so active parameters are far fewer than total.

Q2. Why does Muse Spark Open need about 1 TB of memory?

A2. All 816B parameters must be held in memory even though only 42B are active per token.

Q3. What is the KV cache?

A3. Stored key and value vectors for earlier tokens, so they are not recomputed; it grows with context length.

Q4. Why do agents favour Flash models?

A4. They make many model calls per task, so speed and cost per call dominate.

TOMORROW

What We're Watching: independent benchmarks of Muse Spark Open against Kimi K3 and GLM 5.3. What We're Watching: a MiniMax model card and API price for M3.1 Flash. What We're Watching: real-world KV-cache compression results from DeepSeek's report.

TOPICS COVERED SO FAR (RECAP)

Recent days covered speech-to-speech voice agents, token-efficient Flash models, California's verified-safety and kill-switch plan, open local agent models, Gemini 4 Argon and agent protocols. Today added a frontier-sized open MoE model on Macs, MiniMax's Flash preview and KV-cache compression.

Sources: Engadget, MacRumors, TheNewStack, iClarified and LetsDataScience on Muse Spark Open (engadget.com, macrumors.com, thenewstack.io); eesel.ai and OpenCode data on MiniMax M3.1 Flash (eesel.ai, opencode.ai); DeepSeek V4.1-Flash report (arxiv.org/pdf/2609.19969) and TechNews.tw (technews.tw).