DeepSeek V4: Everything We Know Before It Drops

February 28, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** DeepSeek V4 drops next week (first week of March 2026) with native image and video generation, a 1M token context window, and a rumored 1 trillion MoE parameters with only 32B active per token. If the 32B active count holds, V4 would actually be easier to run per-token than V3 despite being 50% larger. Expected open-weight under Apache 2.0. Huawei gets early access; Nvidia and AMD are locked out. We'll update this guide with GGUF sizes, VRAM tables, and setup instructions the moment weights land.

More on this topic: VRAM Requirements | Best Local Models for OpenClaw | llama.cpp vs Ollama vs vLLM | Fine-Tuning with LoRA and QLoRA

The Financial Times reported on February 27 that DeepSeek will release V4 next week, timed ahead of China's "Two Sessions" parliamentary meetings starting March 4. This is their first major model release since R1 dropped in January 2025 – over a year of silence.

V4 is multimodal from day one. Not text-first with vision bolted on later (the approach most labs take), but native image, video, audio, and text generation built into the architecture. The context window jumps from 128K to 1 million tokens. And based on leaked architecture details, the model may actually be easier to run locally than V3 despite being 50% larger.

Here's everything we know, what we can extrapolate from V3's progression, and how to prepare your hardware before weights drop.

What's confirmed

These come from the Financial Times, Reuters, CNBC, and The Information – multiple outlets with independent sourcing.

Release timing: First week of March 2026. DeepSeek will publish an abbreviated technical note at launch, with a full engineering report following roughly one month later.

Multimodal: Native image, video, audio, and text generation. Not a separate vision encoder strapped to a text model. The architecture handles all modalities from the ground up.

Context window: 1 million tokens, up from 128K in V3/V3.2. DeepSeek quietly upgraded their API to 1M context in mid-February, which the community now reads as a V4 preview.

Huawei optimization: DeepSeek worked with Huawei's Ascend chip division and Chinese AI chipmaker Cambricon to optimize V4 for domestic hardware. More on this below.

Open weights expected: DeepSeek has released every major model under Apache 2.0. They've accumulated 75M+ downloads on Hugging Face. Nothing suggests V4 will break this pattern.

What's rumored

These come from GitHub code leaks, insider reports, and benchmark leaks. Treat them as informed speculation until the technical report lands.

Parameter count: Around 1 trillion total parameters, up from 671B in V3. Still MoE architecture.

Active parameters: Approximately 32B per token, down from 37B in V3. If true, this is the interesting number for local inference: despite being 50% larger overall, V4 would activate fewer parameters per forward pass than V3. Less compute per token means faster generation on the same hardware.

Three architectural innovations (from GitHub leaks and research papers):

Innovation	What it does
Manifold-Constrained Hyper-Connections (mHC)	New layer connectivity pattern replacing standard residual connections
Engram conditional memory	Hash-based lookup tables stored in system RAM (not VRAM), O(1) retrieval regardless of context length. Published January 2026
DeepSeek Sparse Attention (DSA)	Already validated in V3.2. Cuts attention compute overhead by roughly 50%

The Engram memory system is worth paying attention to. If the model stores hash-based lookup tables in system RAM rather than VRAM, it changes the calculus for local inference. Your 64GB of DDR5 becomes part of the model's memory system, not just overflow space for weights that don't fit on the GPU.

MODEL1 GitHub leak: In January 2026, 28 references to an unknown identifier "MODEL1" appeared across 114 files in DeepSeek's FlashMLA repository. The code reveals a distinct

architecture from V3.2 with a restructured KV cache layout, new sparsity handling, FP8 data format changes, and hardware support for both Nvidia Blackwell (SM100) and Hopper GPUs.

Leaked benchmarks (unverified, from internal testing):

Benchmark	V4 (leaked)	Best current
HumanEval	90%	Claude 88%
SWE-bench Verified	>80%	Claude Opus 4.5: 80.9%

Take these with salt. Internal benchmarks often look different from independent evaluation.

The V3 to V4 progression

DeepSeek doesn't do clean version bumps. They iterate rapidly and each point release changes the model meaningfully.

Version	Date	What changed
V3	Dec 2024	671B MoE, 37B active, MLA attention, trained for $5.6M
V3-0324	Mar 2025	Same architecture, better post-training borrowed from R1's RL pipeline
V3.1	Aug 2025	Hybrid thinking mode (reasoning + direct answers in one model)
V3.2	Dec 2025	DeepSeek Sparse Attention, agent-focused training on 85K+ tasks, tool-use reasoning
V4	Mar 2026	1T params, native multimodal, 1M context, Engram memory

The pattern is clear: each release gets bigger but more efficient per token. V3 activated 37B of 671B. V4 reportedly activates 32B of ~1T. The ratio of active-to-total parameters keeps shrinking, which is what makes MoE models increasingly practical for local inference despite growing total size.

The Huawei angle

Reuters reported on February 25 that DeepSeek denied Nvidia and AMD pre-release access to V4 for performance optimization. Huawei's Ascend chip division and other domestic Chinese chipmakers received a multi-week head start instead.

This is the first time a major Chinese AI lab has deliberately locked American chipmakers out of its pre-release pipeline. It breaks a long-standing industry convention where chip vendors get early model access to tune their drivers and software stacks.

What it means for local AI:

The V4 weights themselves will almost certainly work on Nvidia hardware – MoE models are architecture-agnostic at the GGUF/safetensors level. Your RTX 3090 doesn't care who optimized the training run.

But the first-party inference optimizations (FlashMLA kernels, attention implementations, quantization recipes) will target Huawei Ascend first. Nvidia-optimized kernels will come from the community, not from DeepSeek. Expect llama.cpp and vLLM to fill the gap within days of release, the same way they did for V3.

The sanctions backdrop: A senior Trump administration official alleges DeepSeek trained V4 on smuggled Nvidia Blackwell chips clustered at a data center in Inner Mongolia, and plans to publicly claim it used Huawei chips. Neither DeepSeek nor Nvidia have commented. The Huawei Ascend 910C runs at roughly 60% of H100 inference performance – capable for serving, but the training story is murkier.

None of this affects your ability to download and run the weights locally. Open weights are open weights.

Hardware prep: what you'll probably need

We don't have V4 GGUF sizes yet. But we can extrapolate from V3 and the rumored architecture.

V3 baseline (671B total, 37B active)

Quantization	Disk size	VRAM needed	Practical?
Q4_K_M	377 GB	~380-400 GB	Multi-GPU clusters or Mac 512GB
Q2_K (dynamic)	~207 GB	~210 GB	Mac 256GB+ or large multi-GPU

Quantization	Disk size	VRAM needed	Practical?
IQ1_M (1.58-bit)	131 GB	~135 GB	Mac 192GB, barely usable quality

V3 on consumer hardware was brutal. A single RTX 4090 with CPU offloading managed 2-4 tok/s. A Mac Studio M3 Ultra with 512GB unified memory hit about 20 tok/s at Q4. The full 671B model was never practical on anything less than 192GB total memory.

V4 estimates (1T total, ~32B active)

If V4 follows V3's MoE structure with 50% more total parameters:

Quantization	Estimated disk size	Estimated memory needed
Q4_K_M	~550-600 GB	~570-620 GB
Q2_K (dynamic)	~300-330 GB	~320-350 GB
IQ1_M (1.58-bit)	~190-210 GB	~200-220 GB

The silver lining: 32B active params vs V3's 37B means generation speed should be slightly faster per token on the same hardware, even though you need more memory to load the full model. MoE routing only touches a fraction of the weights each token.

What hardware can realistically run V4

Setup	Can it run V4?	Expected speed
Mac Studio M4 Ultra 512GB	Yes, Q4	~15-25 tok/s (estimated)
Mac Studio M4 Ultra 256GB	Yes, at IQ2 or lower	~8-15 tok/s
Mac Mini cluster (8x 64GB via exo)	Maybe, at Q4	Depends on interconnect
4x RTX 4090 (96GB) + 256GB RAM	Partial GPU offload	~3-8 tok/s
Single RTX 4090/5090 + CPU offload	Technically yes	~1-3 tok/s
RTX 3090 single card	No (full model)	Use distilled versions

The real play for most people: Wait for distilled versions. DeepSeek released R1-Distill models from 1.5B to 70B, all of which run on consumer hardware. V4 distills will almost certainly follow, and a V4-Distill-32B at Q4 would fit on a single RTX 3090 at 30+ tok/s.

How to prepare your setup now

You can do all of this today so you're ready the moment weights drop.

Update your inference stack

```
# llama.cpp — get the latest build with MLA support
cd llama.cpp && git pull && cmake -B build -DGGML_CUDA=ON && cmake --build build -j

# Ollama — update to latest
curl -fsSL https://ollama.ai/install.sh | sh

# vLLM — update
pip install -U vllm
```

DeepSeek V3/R1 support has been in llama.cpp since January 2025. V4 will likely need new architecture support (Engram memory, mHC connections), but the MoE + MLA foundation is already there. Expect community PRs within hours of weight release.

Check your VRAM headroom

```
# See what's using your GPU memory right now
nvidia-smi

# Check total system RAM (you'll need this for CPU offload)
free -h
```

For the full V4 model, you need total memory (VRAM + RAM) in the 200-600 GB range depending on quantization. For distilled versions, a 32B distill at Q4 needs about 20 GB.

Clear disk space

V3's Q4_K_M GGUF was 377 GB across 9 split files. V4 will be bigger. Make sure you have at least 600 GB free for the download plus working space.

```
# Check available disk space
df -h /path/to/your/models
```

Bookmark the sources

When V4 drops, GGUFs will appear on Hugging Face from these uploaders first:

- unsloth – dynamic quantizations, usually fastest
- bartowski – standard imatrix-calibrated GGUFs
- lmstudio-community – LM Studio compatible

The official weights will be at deepseek-ai on Hugging Face in safetensors format.

Know the gotchas from V3

DeepSeek models in llama.cpp have quirks. Save yourself the debugging:

- **MLA is required.** Recent DeepSeek GGUFs need MLA-aware llama.cpp builds. Older builds won't load the model at all.
- **Split file handling.** Models this size come as multiple GGUF parts (V3 was 9 parts). Some tools need you to merge first; llama.cpp handles splits natively.
- **Partial GPU offload is flakey.** Offloading some-but-not-all layers with MLA enabled has issues in mainline llama.cpp. The ik_llama.cpp fork handles this better.
- **Chat template errors.** llama.cpp's jinja renderer chokes on DeepSeek's `.split()` usage. You may need `--override-kv` or a custom template.
- **Dynamic quants + tool calling crash.** Unsloth's UD-GGUF quants crash when using function calling in llama.cpp.

What to run while you wait

If you want to get familiar with DeepSeek's architecture before V4 drops, these run on consumer hardware today:

Model	What it is	VRAM needed (Q4)	Ollama tag
R1-Distill-Qwen-7B	Reasoning distill, fits anywhere	6 GB	`deepseek-r1:7b`

Model	What it is	VRAM needed (Q4)	Ollama tag
R1-Distill-Qwen-14B	Better reasoning, still small	10 GB	`deepseek-r1:14b`
R1-Distill-Qwen-32B	Best distill for single GPU	20 GB	`deepseek-r1:32b`
R1-Distill-Llama-70B	Top distill, needs 48GB+	40 GB	`deepseek-r1:70b`
DeepSeek-V3.2 (full)	685B MoE, needs serious hardware	~400 GB	`deepseek-v3.2`

The distilled models are standard Qwen/Llama architectures – no MoE, no MLA, no special handling needed. They just work in Ollama, llama.cpp, LM Studio, and vLLM. The R1-Distill-Qwen-32B in particular is worth running: it fits on a single RTX 3090 at Q4 and outperforms OpenAI's o1-mini on math reasoning.

```
# Try the 32B reasoning distill right now
ollama pull deepseek-r1:32b
ollama run deepseek-r1:32b
```

Post-launch update sections

We'll fill these in the moment V4 weights are available.

VRAM requirements by quantization

Coming soon. GGUF sizes and memory requirements for every quant level.

Speed benchmarks by hardware tier

Coming soon. Generation speed on RTX 3090, 4090, 5090, Mac M-series, and multi-GPU setups.

Quantization recommendations

Coming soon. Which quant to pick for your hardware, and any Unsloth dynamic quant issues to watch for.

Setup guide

Coming soon. Step-by-step for Ollama, llama.cpp, vLLM, and MLX.

Distilled model lineup

Coming soon. Sizes, benchmarks, and VRAM requirements for V4 distills once released.

Bottom line

The 1T parameter count sounds scary, but if the 32B active-parameter rumor holds, V4 is actually a step forward for local inference: fewer active params than V3, with better quality per token. Native image and video generation from an open-weight model is something no one else has shipped yet.

For most people on consumer hardware, the play is the same as V3: wait for distilled versions. A V4-Distill-32B at Q4 would fit on a single 24GB card with room for context. If you're on a Mac with 192GB+ unified memory or running a multi-GPU setup, the full model at aggressive quantization becomes interesting.

Update your stack, clear your disk, and bookmark this page. We'll update it the moment weights drop.
