FLASHML • FREETOKEN

Модель на 284B на игровом ПК

DeepSeek V4 Flash

Model: DeepSeek V4 Flash
Params: 284B
Backend: Local (RTX 5090)
Status: Running ✓

deepseek

DeepSeek V4 Flash

Запускается локально
на одной RTX 5090

284B

FlashML • FreeToken

FreeToken

Запускай 290B+ MoE модели локально на своём ПК

Большие модели (290B+ MoE)

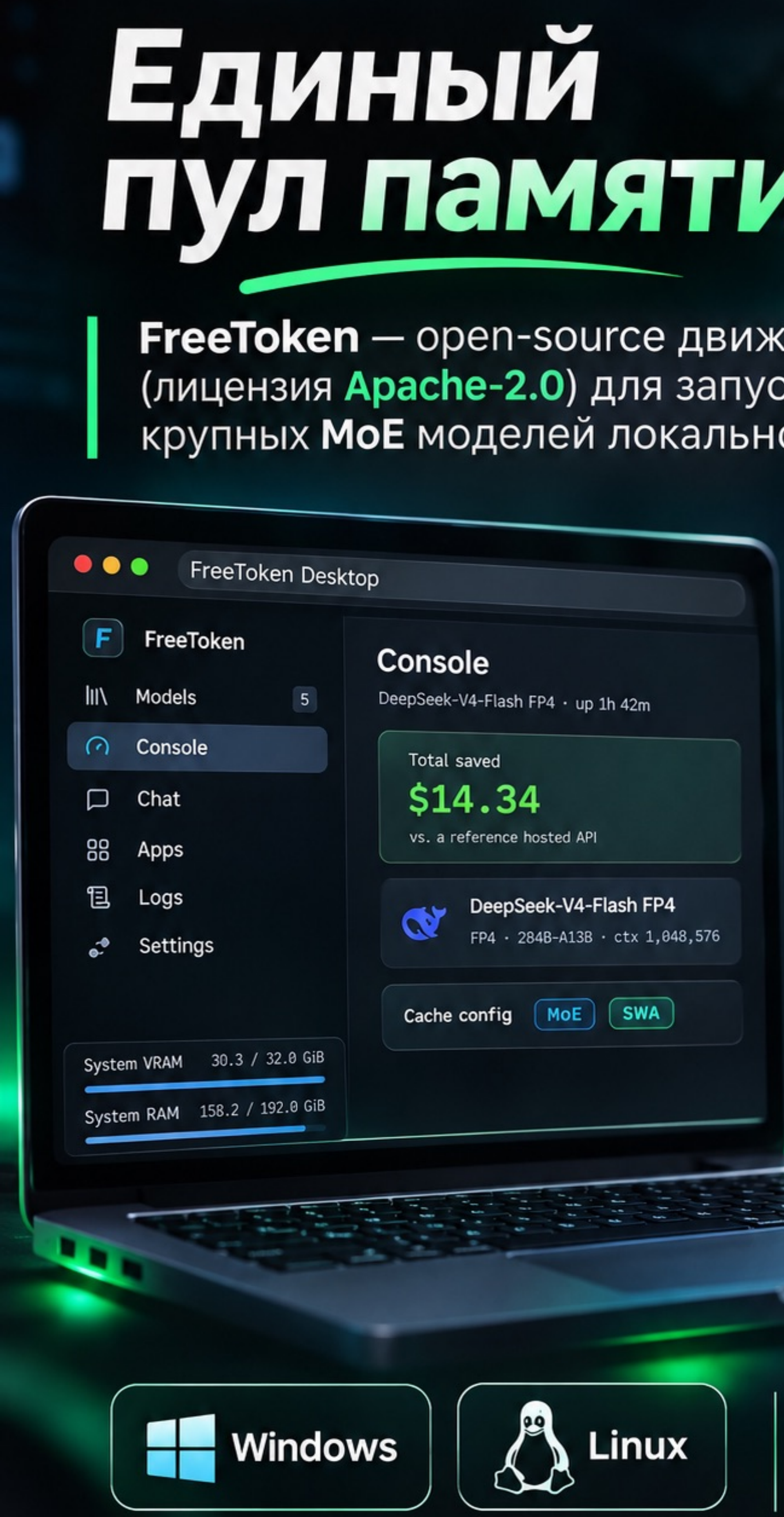
Локальный запуск (на своём ПК)

Быстро и эффективно

FreeToken • FlashML

Единый пул памяти

FreeToken — open-source движок (лицензия **Apache-2.0**) для запуска крупных **MoE** моделей локально.



Объединяет память **GPU, CPU** и **System RAM**

Позволяет запускать модели, значительно больше объёма вашей VRAM

Создан командой разработчиков **Apache Spark**

Windows | Linux | Open Source (Apache-2.0)

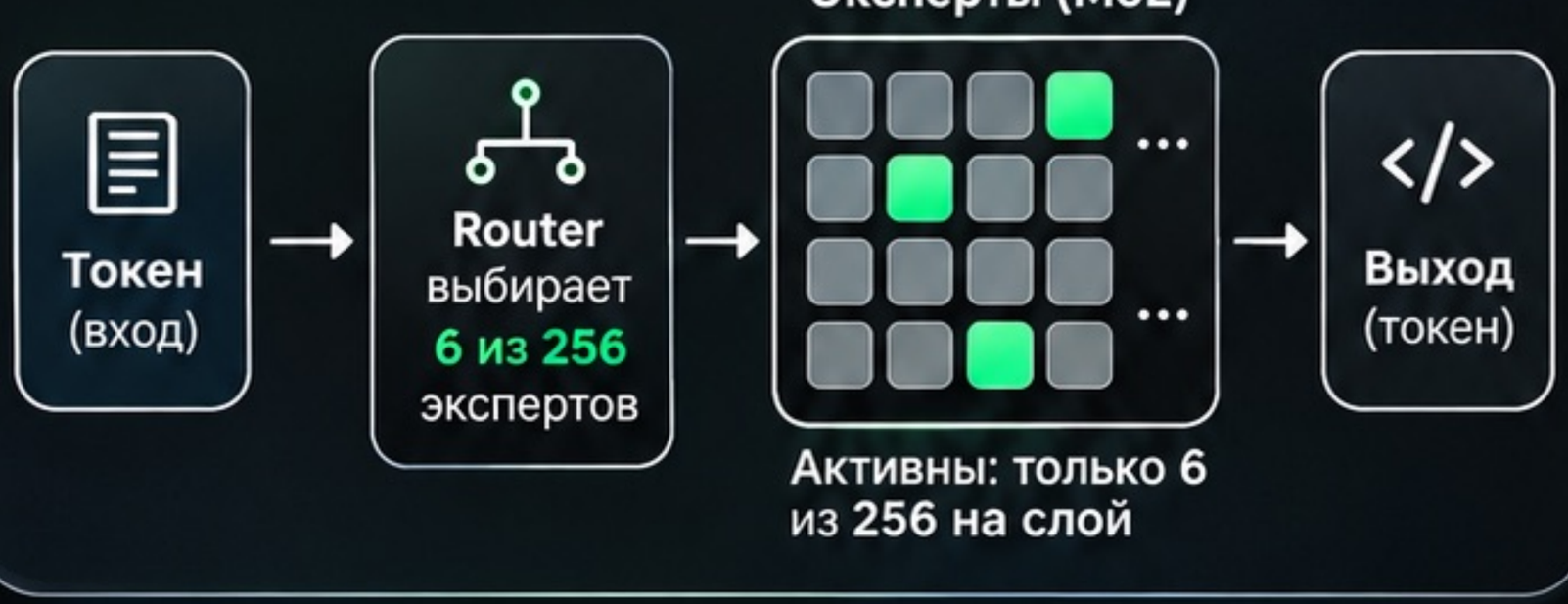
DeepSeek V4 Flash

Как работает 13B из 284B?

Модель DeepSeek-V4-Flash использует архитектуру **MoE**, где из **284B** параметров на каждый токен активируются только **~13B**.



MoE (Mixture-of-Experts)



Как это работает?

Горячие эксперты хранятся в **VRAM (GPU)**, остальные — в **SystemRAM** и подгружаются по запросу (**streaming**).

284B
общий размер модели

~13B
активных параметров на токен

6 из 256
экспертов на слой

43
слоя (layers)

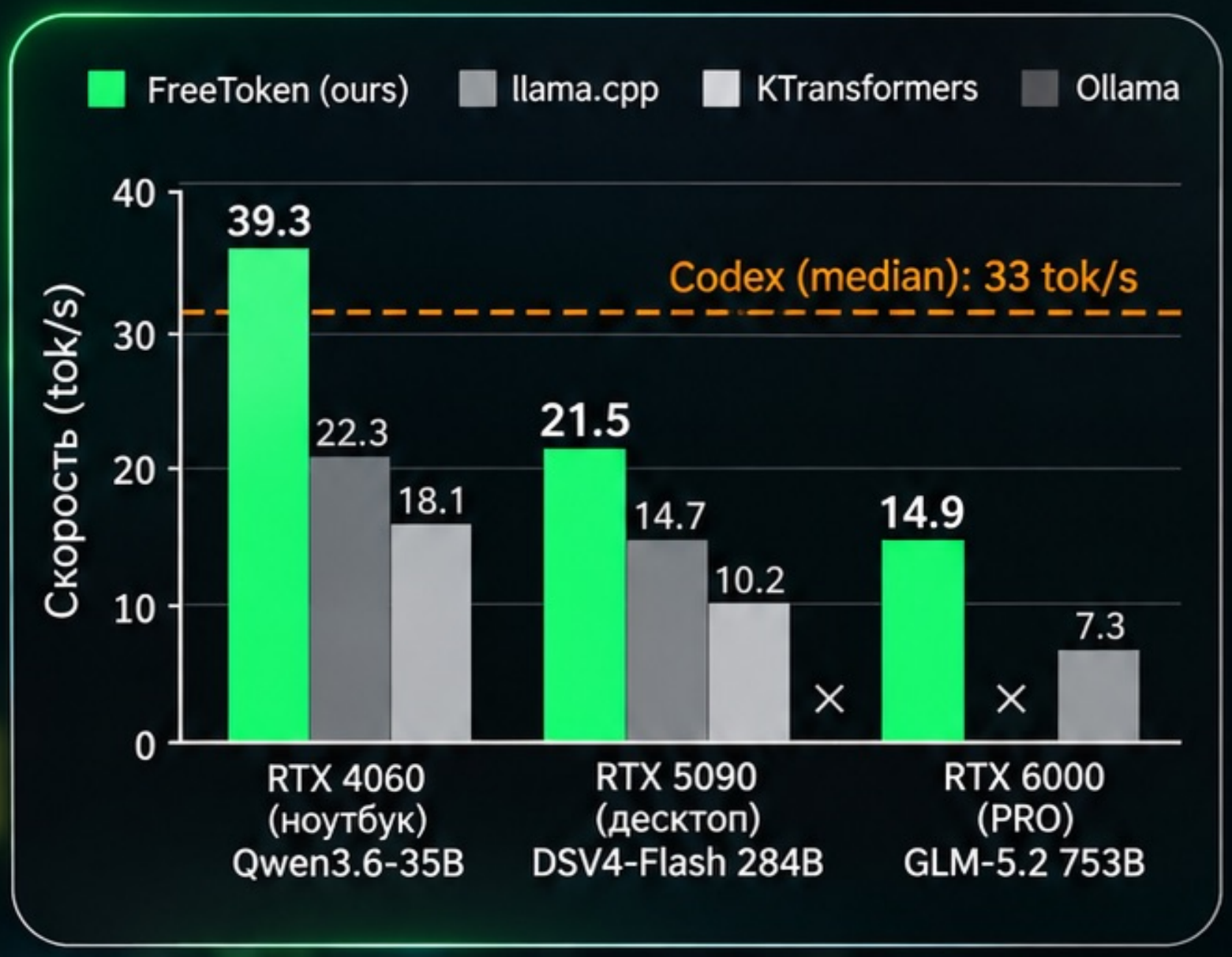


FreeToken • DeepSeek V4 Flash

Скорость: 21.5 tok/s

Модель на **284B параметров** работает с **~21.5 tok/s** на десктопе с одной **RTX 5090** (тест **coding-agent**).





284B параметров

21.5 tok/s на RTX 5090

Тест: **coding-agent**

llama.cpp — 14.7
KTransformers — 10.2
Ollama — не работает

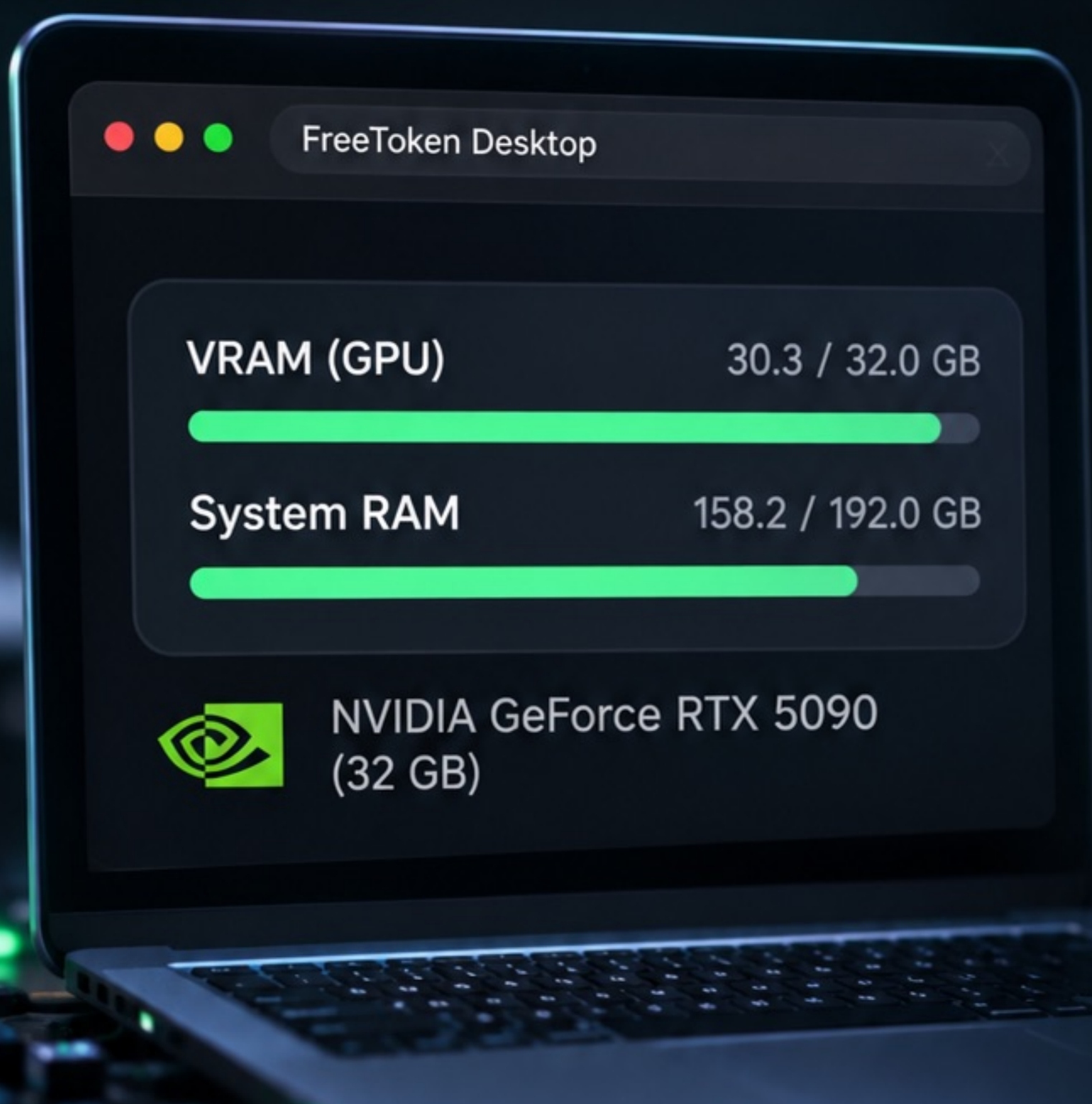
Для **зравнения:** ноутбук с **RTX 4060** (8GB) запускает **35B модель** с **39.3 tok/s.**

FreeToken • DeepSeek V4 Flash

Нужно 192 GB RAM

Пример: десктоп с **RTX 5090** (32 GB) и **192 GB DDR5** успешно запускает модель локально.





192 GB DDR5
для хранения экспертов (MoE)

Требуется **NVIDIA RTX 30 / 40 / 50**

Версии для **macOS** нет

Поддержка **AMD** в разработке

Эксперты (MoE) хранятся в **System RAM**, поэтому важен объём **RAM**, а **не VRAM**.

FreeToken • DeepSeek V4 Flash

Установка в один pip

Установи, укажи модель с Hugging Face и запусти локальный OpenAI/Anthropic-совместимый API (порт 1919).

Terminal

```
# Создать окружение
$ uv venv && source .venv/bin/activate

# Установить FreeToken с ускорением
$ uv pip install "freetoken[accel]"

# Запустить модель (порт 1919)
$ ft serve --model ~/models/Qwen3.6-35B-A3B

# Подключить в свой coding-agent
$ ft launch claude
```

Hugging Face

Совместимый API с **OpenAI** и **Anthropic**

Работает с **Claude Code, Codex, OpenCode** и **Hermes**

Локальный запуск вашего **LLM**

Требования | Linux CLI | CUDA 13+ | Driver r580+