Hardware support

- GPU Selection
- Linux Suspend Resume
- AMD Radeon
  - Linux Support
  - Windows Support
  - Overrides on Linux
  - GPU Selection
  - Container Permission
- Metal (Apple GPUs)
- Vulkan GPU Support
  - GPU Selection

Ollama supports Nvidia GPUs with compute capability 5.0+ and driver version 550 and newer. Nvidia GPUs with compute capability 5.0 through 6.2 require driver version 570 or newer.

Check your compute compatibility to see if your card is supported:

https://developer.nvidia.com/cuda-gpus

Compute Capability	Family	Cards
12.1	NVIDIA	`GB10 (DGX Spark)`
12.0	GeForce RTX 50xx	`RTX 5060` `RTX 5060 Ti` `RTX 5070` `RTX 5070 Ti` `RTX 5080` `RTX 5090`
	NVIDIA Professional	`RTX PRO 4000 Blackwell` `RTX PRO 4500 Blackwell` `RTX PRO 5000 Blackwell` `RTX PRO 6000 Blackwell`
9.0	NVIDIA	`H200` `H100`
8.9	GeForce RTX 40xx	`RTX 4090` `RTX 4080 SUPER` `RTX 4080` `RTX 4070 Ti SUPER` `RTX 4070 Ti` `RTX 4070 SUPER` `RTX 4070` `RTX 4060 Ti` `RTX 4060`
	NVIDIA Professional	`L4` `L40` `RTX 6000`

Compute Capability	Family	Cards
8.6	GeForce RTX 30xx	`RTX 3090 Ti` `RTX 3090` `RTX 3080 Ti` `RTX 3080` `RTX 3070 Ti` `RTX 3070` `RTX 3060 Ti` `RTX 3060` `RTX 3050 Ti` `RTX 3050`
	NVIDIA Professional	`A40` `RTX A6000` `RTX A5000` `RTX A4000` `RTX A3000` `RTX A2000` `A10` `A16` `A2`
8.0	NVIDIA	`A100` `A30`
7.5	GeForce GTX/RTX	`GTX 1650 Ti` `TITAN RTX` `RTX 2080 Ti` `RTX 2080` `RTX 2070` `RTX 2060`
	NVIDIA Professional	`T4` `RTX 5000` `RTX 4000` `RTX 3000` `T2000` `T1200` `T1000` `T600` `T500`
	Quadro	`RTX 8000` `RTX 6000` `RTX 5000` `RTX 4000`
7.0	NVIDIA	`TITAN V` `V100` `Quadro GV100`
6.1	NVIDIA TITAN	`TITAN Xp` `TITAN X`
	GeForce GTX	`GTX 1080 Ti` `GTX 1080` `GTX 1070 Ti` `GTX 1070` `GTX 1060` `GTX 1050 Ti` `GTX 1050`
	Quadro	`P6000` `P5200` `P4200` `P3200` `P5000` `P4000` `P3000` `P2200` `P2000` `P1000` `P620` `P600` `P500` `P520`
	Tesla	`P40` `P4`
6.0	NVIDIA	`Tesla P100` `Quadro GP100`
5.2	GeForce GTX	`GTX TITAN X` `GTX 980 Ti` `GTX 980` `GTX 970` `GTX 960` `GTX 950`
	Quadro	`M6000 24GB` `M6000` `M5000` `M5500M` `M4000` `M2200` `M2000` `M620`
	Tesla	`M60` `M40`
5.0	GeForce GTX	`GTX 750 Ti` `GTX 750` `NVS 810`
	Quadro	`K2200` `K1200` `K620` `M1200` `M520` `M5000M` `M4000M` `M3000M` `M2000M` `M1000M` `K620M` `M600M` `M500M`

For building locally to support older GPUs, see development.

GPU Selection

If you have multiple NVIDIA GPUs in your system and want to limit Ollama to use a subset, you can set `CUDA_VISIBLE_DEVICES` to a comma separated list of GPUs. Numeric IDs may be used, however

ordering may vary, so UUIDs are more reliable. You can discover the UUID of your GPUs by running `nvidia-smi -L` If you want to ignore the GPUs and force CPU usage, use an invalid GPU ID (e.g., "-1")

Linux Suspend Resume

On linux, after a suspend/resume cycle, sometimes Ollama will fail to discover your NVIDIA GPU, and fallback to running on the CPU. You can workaround this driver bug by reloading the NVIDIA UVM driver with `sudo rmmod nvidia_uvm && sudo modprobe nvidia_uvm`

AMD Radeon

Ollama supports the following AMD GPUs via the ROCm library:

> **NOTE:** Additional AMD GPU support is provided by the Vulkan Library - see below.

Linux Support

Ollama requires the AMD ROCm v7 driver on Linux. You can install or upgrade using the `amdgpu-install` utility from AMD's ROCm documentation.

Family	Cards and accelerators
AMD Radeon RX	`9070 XT` `9070 GRE` `9070` `9060 XT` `9060 XT LP` `9060` `7900 XTX` `7900 XT` `7900 GRE` `7800 XT` `7700 XT` `7700` `7600 XT` `7600` `6950 XT` `6900 XTX` `6900XT` `6800 XT` `6800`
AMD Radeon AI PRO	`R9700` `R9600D`
AMD Radeon PRO	`W7900` `W7800` `W7700` `W7600` `W7500` `W6900X` `W6800X Duo` `W6800X` `W6800` `V620`
AMD Ryzen AI	`Ryzen AI Max+ 395` `Ryzen AI Max 390` `Ryzen AI Max 385` `Ryzen AI 9 HX 475` `Ryzen AI 9 HX 470` `Ryzen AI 9 465` `Ryzen AI 9 HX 375` `Ryzen AI 9 HX 370` `Ryzen AI 9 365`
AMD Instinct	`MI350X` `MI300X` `MI300A` `MI250X` `MI250` `MI210` `MI100`

Windows Support

Ollama requires an AMD ROCm v7 / HIP7-capable driver stack on Windows.

Family	Cards and accelerators
AMD Radeon RX	`7900 XTX` `7900 XT` `7900 GRE` `7800 XT` `7700 XT` `7600 XT` `7600`
AMD Radeon PRO	`W7900` `W7800` `W7700` `W7600` `W7500`

Overrides on Linux

Ollama leverages the AMD ROCm library, which does not support all AMD GPUs. In some cases you can force the system to try to use a similar LLVM target that is close. For example The Radeon RX 5400 is `gfx1034` (also known as 10.3.4) however, ROCm does not currently support this target. The closest support is `gfx1030`. You can use the environment variable `HSA_OVERRIDE_GFX_VERSION` with `x.y.z` syntax. So for example, to force the system to run on the RX 5400, you would set `HSA_OVERRIDE_GFX_VERSION="10.3.0"` as an environment variable for the server. If you have an unsupported AMD GPU you can experiment using the list of supported types below.

If you have multiple GPUs with different GFX versions, append the numeric device number to the environment variable to set them individually. For example, `HSA_OVERRIDE_GFX_VERSION_0=10.3.0` and `HSA_OVERRIDE_GFX_VERSION_1=11.0.0`

At this time, the known supported GPU types on linux are the following LLVM Targets. This table shows some example GPUs that map to these LLVM targets:

LLVM Target	An Example GPU
gfx908	Radeon Instinct MI100
gfx90a	Radeon Instinct MI210/MI250
gfx942	Radeon Instinct MI300X/MI300A
gfx950	Radeon Instinct MI350X
gfx1030	Radeon PRO V620
gfx1100	Radeon PRO W7900
gfx1101	Radeon PRO W7700
gfx1102	Radeon RX 7600
gfx1150	Ryzen AI 9 HX 375
gfx1151	Ryzen AI Max+ 395
gfx1200	Radeon RX 9070
gfx1201	Radeon RX 9070 XT

Reach out on Discord or file an issue for additional help.

GPU Selection

If you have multiple AMD GPUs in your system and want to limit Ollama to use a subset, you can set `ROCR_VISIBLE_DEVICES` to a comma separated list of GPUs. You can see the list of devices with `rocminfo`. If you want to ignore the GPUs and force CPU usage, use an invalid GPU ID (e.g., "-1"). When available, use the `Uuid` to uniquely identify the device instead of numeric value.

Container Permission

In some Linux distributions, SELinux can prevent containers from accessing the AMD GPU devices. On the host system you can run `sudo setsebool container_use_devices=1` to allow containers to use devices.

Metal (Apple GPUs)

Ollama supports GPU acceleration on Apple devices via the Metal API.

Vulkan GPU Support

Additional GPU support on Windows and Linux is provided via Vulkan. Vulkan is enabled by default when the backend is installed. On Windows most GPU vendors drivers come bundled with Vulkan support and require no additional setup steps. Most Linux distributions require installing additional components, and you may have multiple options for Vulkan drivers between Mesa and GPU Vendor specific packages

- Linux Intel GPU Instructions - https://dgpu-docs.intel.com/driver/client/overview.html
- Linux AMD GPU Instructions - https://amdgpu-install.readthedocs.io/en/latest/install-script.html#specifying-a-vulkan-implementation

For AMD GPUs on some Linux distributions, you may need to add the `ollama` user to the `render` group.

The Ollama scheduler leverages available VRAM data reported by the GPU libraries to make optimal scheduling decisions. Vulkan requires additional capabilities or running as root to expose this available VRAM data. If neither root access or this capability are granted, Ollama will use approximate sizes of the models to make best effort scheduling decisions.

```
sudo setcap cap_perfmon+ep /usr/local/bin/ollama
```

GPU Selection

To select specific Vulkan GPU(s), you can set the environment variable `GGML_VK_VISIBLE_DEVICES` to one or more numeric IDs on the Ollama server as described in the FAQ. If you encounter any problems with Vulkan based GPUs, you can disable all Vulkan GPUs by setting `OLLAMA_VULKAN=0` or `GGML_VK_VISIBLE_DEVICES=-1`.

On mixed iGPU/dGPU systems where the Vulkan iGPU is unstable, keep Vulkan enabled and set `GGML_VK_VISIBLE_DEVICES` to the discrete GPU index. For example, use `GGML_VK_VISIBLE_DEVICES=1` when `Vulkan1` is the discrete GPU.

□□□ #2
□ □□□ □□□ 2026-10-31 13:17:17 UTC
□ □□□ □□□ 2026-10-31 13:17:18 UTC