

Docker on different OSs



does the way Docker run on Windows and macOS differ from the way it runs on Linux?

Yes — the way Docker runs on **Windows** and **macOS** is **fundamentally different** from how it runs on **Linux**. Here's the breakdown:

On Linux

- Docker runs **natively** because Linux provides the kernel features Docker depends on:
 - **Namespaces** (isolation)
 - **Control groups (cgroups)** (resource limits)
 - **Union file systems** (like overlayfs, AUFS, etc.)
- The Docker daemon (**dockerd**) directly interacts with the **Linux kernel** — no virtualization layer needed.
- Containers on Linux are **lightweight processes**, not virtual machines.

☐ On Windows

- Two different modes:
 1. **Windows containers** – use Windows kernel isolation (for Windows-based images).
 - These run **natively on Windows Server** or **Windows 10/11 Pro/Enterprise** with container support.
 2. **Linux containers** – need a **Linux kernel**, which Windows doesn't have.
 - Docker Desktop for Windows runs a **lightweight Linux VM** using **Hyper-V** or **WSL2** (Windows Subsystem for Linux v2).
 - Docker daemon and containers live **inside that VM**; Docker CLI on Windows communicates with it over a socket.

🍏 On macOS

- macOS doesn't use the Linux kernel either.
- Docker Desktop runs a **Linux virtual machine** (using **Apple Hypervisor.framework** or **QEMU** on older Intel Macs).
- The Docker engine and containers live **inside that Linux VM**.
- Like Windows, the Docker CLI on macOS talks to the daemon via a network socket inside the VM.

⚙️ Summary Table

Platform	Native or VM	Kernel Used	Notes
Linux	Native	Linux	Docker runs directly on the host kernel
Windows (Linux containers)	VM (Hyper-V or WSL2)	Linux	Docker Desktop uses a Linux VM
Windows (Windows containers)	Native	Windows	Runs only Windows-based images
macOS	VM (Hypervisor.framework/QEMU)	Linux	Docker Desktop runs Linux VM under the hood

So:

- **Linux** → **no virtualization** (native)
- **Windows/macOS** → **needs a Linux VM** to emulate the kernel

Would you like me to show how to verify which kernel or environment your Docker daemon is actually running under on each OS?