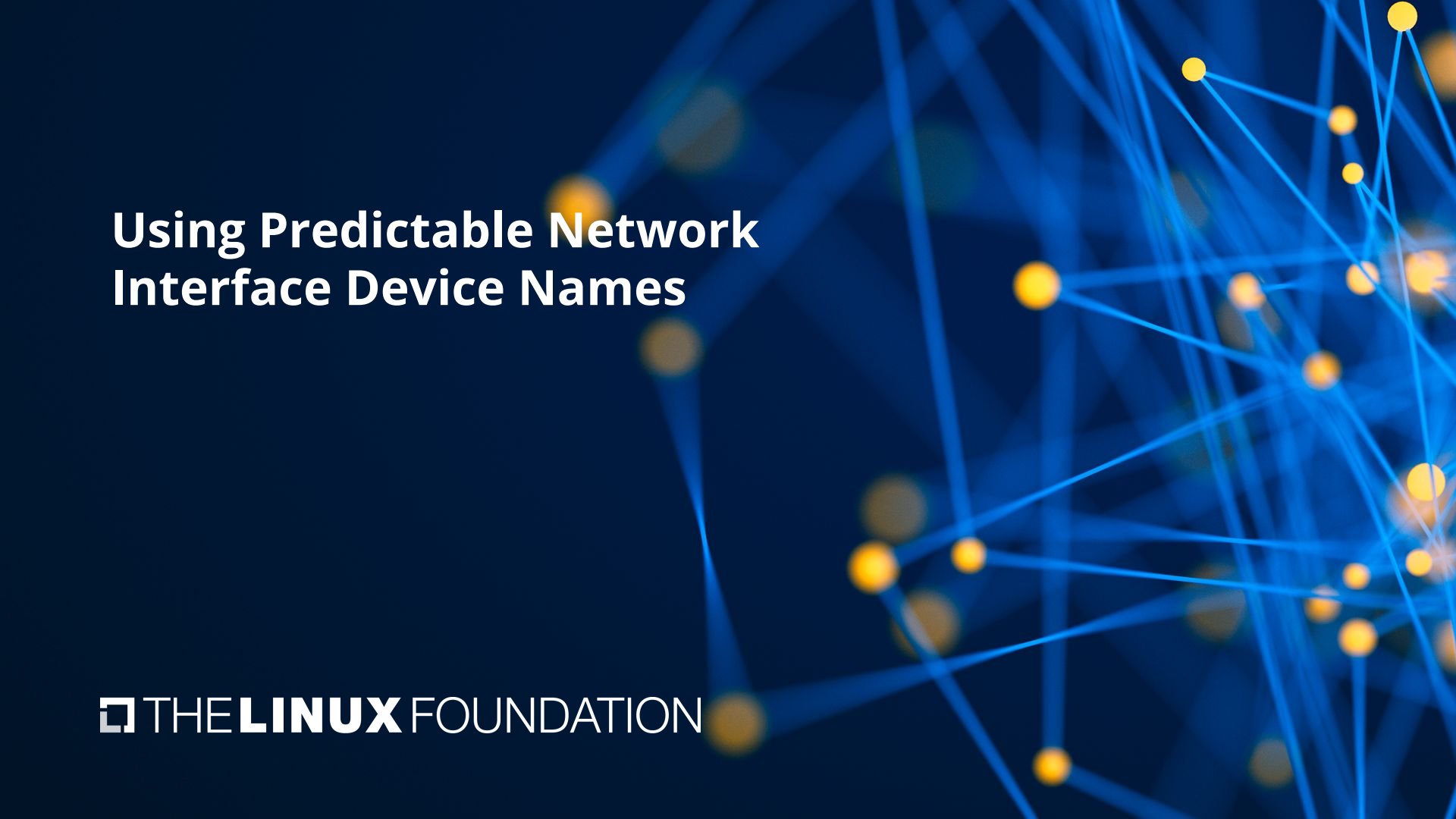
Using Predictable Network Interface Device Names



THE LINUX FOUNDATION

Naming Networks

- Simply naming network devices as **eth0**, **eth1**, etc. can be problematic when multiple interfaces exist, or when the order in which the system probes for and then finds them is not deterministic and can depend on kernel version and options
- Many system administrators have solved this problem in a simple manner, by hard-coding associations between hardware (MAC) addresses and device names in system configuration files and startup scripts
- A more modern approach is offered by the **Predictable Network Interface Device Names** scheme, which is strongly correlated with the use of udev and integration with systemd

Five Types of Names

- There are now 5 types of names that devices can be given:
  - Incorporating Firmware or BIOS provided index numbers for on-board devices, e.g. **eno1**
  - Incorporating Firmware or BIOS provided PCI Express hotplug slot index numbers, e.g. **ens1**
  - Incorporating physical and/or geographical location of the hardware connection, e.g. **enp2s0**
  - Incorporating the MAC address, e.g. **enx7837d1ea46da**
  - Using the old classic method, e.g. **eth0**
- On a machine with two onboard PCI network interfaces that would have been **eth0** and **eth1** in the older naming system:

```
$ ifconfig | grep enp
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
enp4s2: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
```

- It is easy to turn off the new scheme and go back to the classic names, if so desired