Slides: LA-UR-21-24529
Video: LA-UR-21-24530

More Performant Cluster State Management

Using Open Source Firmware and a Kraken

Devon T. Bautista, J. Lowell Wofford *Los Alamos National Laboratory* **01 June 2021**

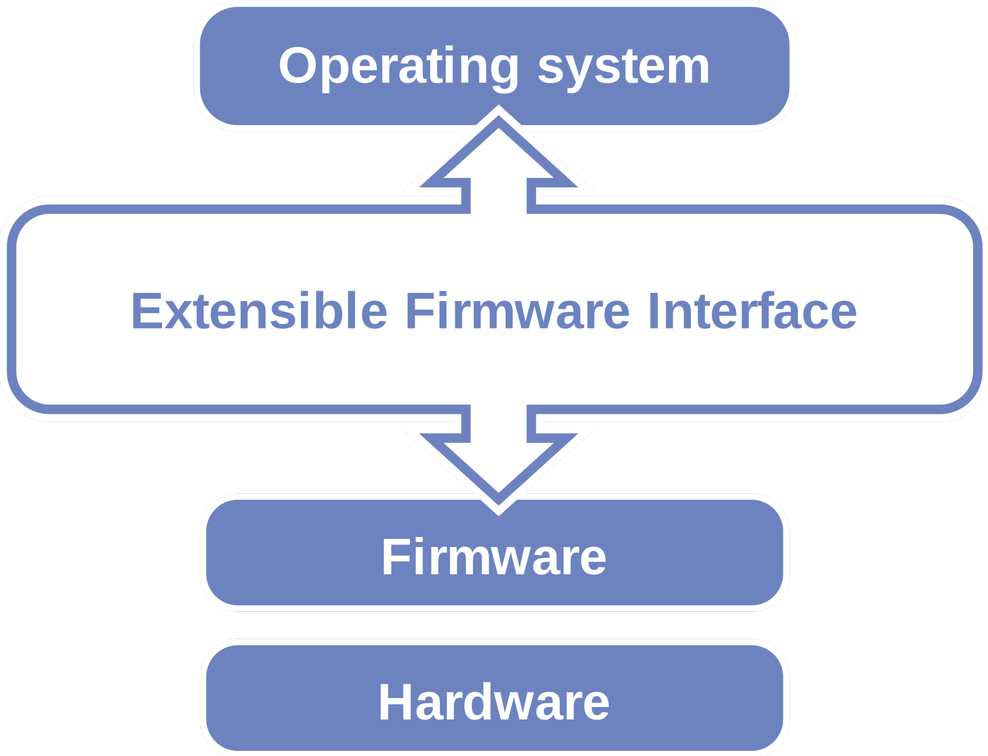
A Little Background on Firmware

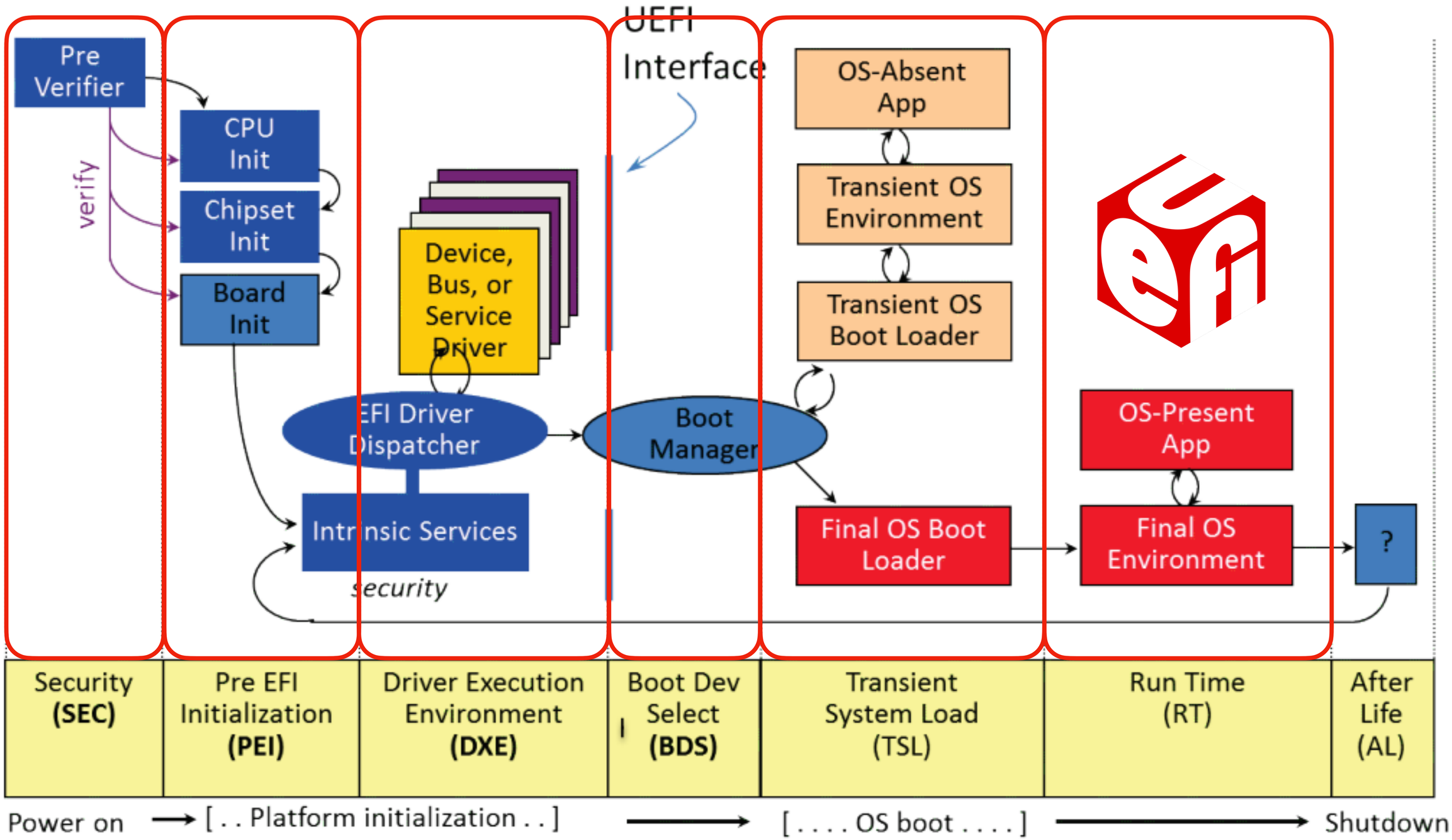
UEFI: How We Got Here

BIOS mechanism:

- *Mundane*
  - Control given to first boot loader found
  - Blindly executes
- *Proprietary*
  - Sans Libreboot, Coreboot
- *Limited*
  - 16-bit real-mode addressing
  - Operates in up to 1 MB of space
  - Programmed in assembly language
  - Can only address up to 2.2 TB drives

- 1998: “Extensible Firmware Interface”
- 2005: “*Unified* Extensible Firmware Interface”




UEFI Interface
Pre Verifier
verify
CPU Init
Chipset Init
Board Init
Device, Bus, or Service Driver
EFI Driver Dispatcher
Intrinsic Services
security
Boot Manager
OS-Absent App
Transient OS Environment
Transient OS Boot Loader
Final OS Boot Loader
OS-Present App
Final OS Environment
?
Security (SEC)
Pre EFI Initialization (PEI)
Driver Execution Environment (DXE)
Boot Dev Select (BDS)
Transient System Load (TSL)
Run Time (RT)
After Life (AL)
Power on
[. . Platform initialization . .]
[. . . . OS boot . . . .]
Shutdown

Towards Open Source Firmware

UefiOpromSetup

OemBadgingDxe
JpegDecoderDxe
ItkLogoProcess

MsftOemActivation

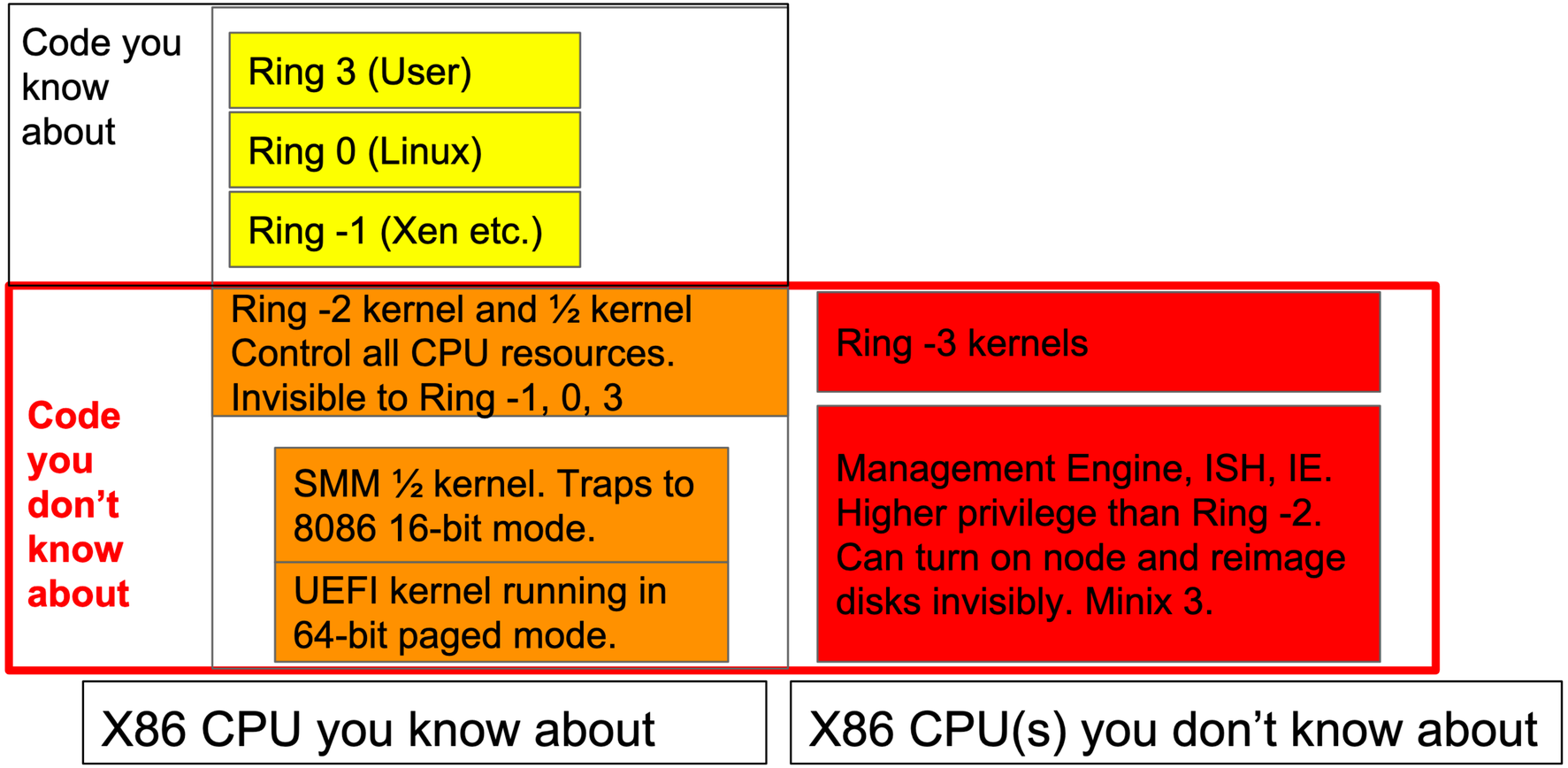
VlanConfigDxe
ArpDxe
Dhcp4Dxe
Ip4Dxe
Mtftp4Dxe
Udp4Dxe
Tcp4Dxe
UefiPxeBcDxe
IScsiDxe
Ip6Dxe
TcpDxe
Udp6Dxe
Dhcp6Dxe
Mtftp6Dxe
HttpDxe
HttpBootDxe
HttpUtilitiesDxe
DnsDxe

IsaBusDxe
IsaFloppyDxe
IsaSerialDxe
LegacyRegion2
LegacyBiosPlatform
Legacy8259
LegacyInterrupt
LegacyInterruptHookDxe
LegacyBiosDxe

Y2KRollover

[1] [2]

The 2½ “Hidden OSes” on x86



[3]

Redundant Drivers

	EDK2 Firmware	GRUB Bootloader	Linux Kernel
Filesystem	tianocore / edk2 Code · Pull requests 5 · Actions · Projects master · edk2 / FatPkg / EnhancedFatDxe /	git index : grub.git GNU GRUB summary · refs · log · tree · commit · diff path: root/grub-core/fs/fat.c	/ fs / fat.h /* SPDX-License-Identifier: GPL-2.0 */ #ifndef _FAT_H #define _FAT_H #include <linux/buffer_head.h> #include <linux/nls.h>
USB	tianocore / edk2 Code · Pull requests 5 · Actions · Projects · Security · Insigl master · edk2 / MdeModulePkg / Bus / Usb / UsbBusDxe / UsbBus.c	git index : grub.git GNU GRUB summary · refs · log · tree · commit · diff path: root/grub-core/bus/usb/usb.c	/ drivers / usb / core / usb.c // SPDX-License-Identifier: GPL-2.0 /* * drivers/usb/core/usb.c * * (C) Copyright Linus Torvalds 1999 * (C) Copyright Johannes Erdfelt 1999-2001
Network	tianocore / edk2 Code · Pull requests 5 · Actions · Proje master · edk2 / NetworkPkg / TcpDxe /	git index : grub.git GNU GRUB · grub.git summary · refs · log · tree · commit · diff path: root/grub-core/net/tcp.c	/ net / ipv4 / tcp.c // SPDX-License-Identifier: GPL-2.0-or-later /* * INET An implementation of the TCP * operating system. INET is i * interface as the means of cc *

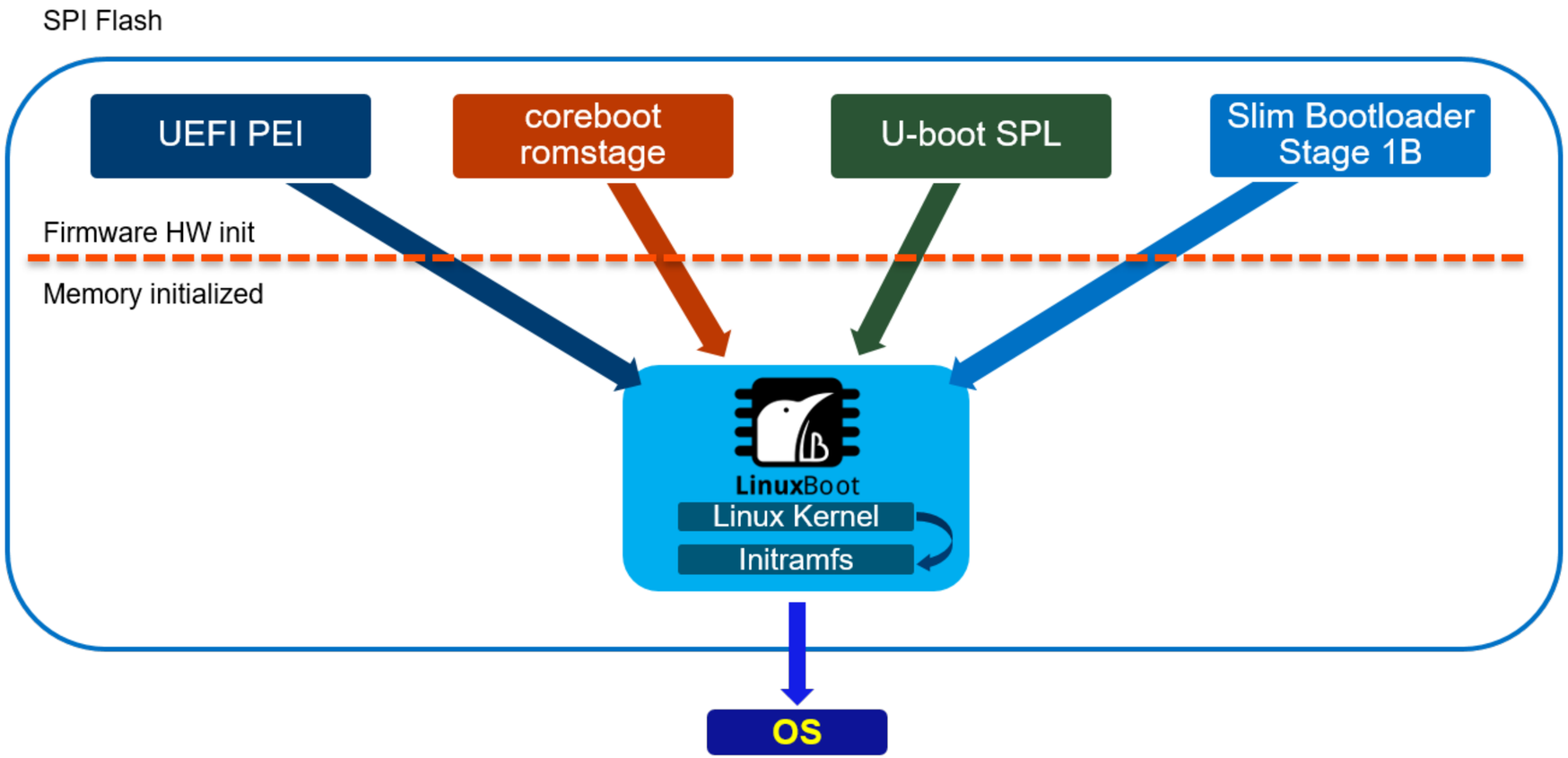
Problems

- Redundant drivers with little-to-no code sharing
  - Increased attack surface
  - Loading the same drivers multiple times is slow
- The code with the highest privilege and control over hardware is audited the least
  - Less frequent update/deployment lifecycle than most software and operating systems
  - Proprietary, closed-source
- Reliance on vendor for updates and fixes
  - A bottleneck to your production timeline
  - Outsourcing development to *another* middleman in mitigating firmware updates/issues

Scary! What is being done?

“Let Linux do it.”

Replacing redundant, closed drivers with vetted, open ones



https://www.linuxboot.org

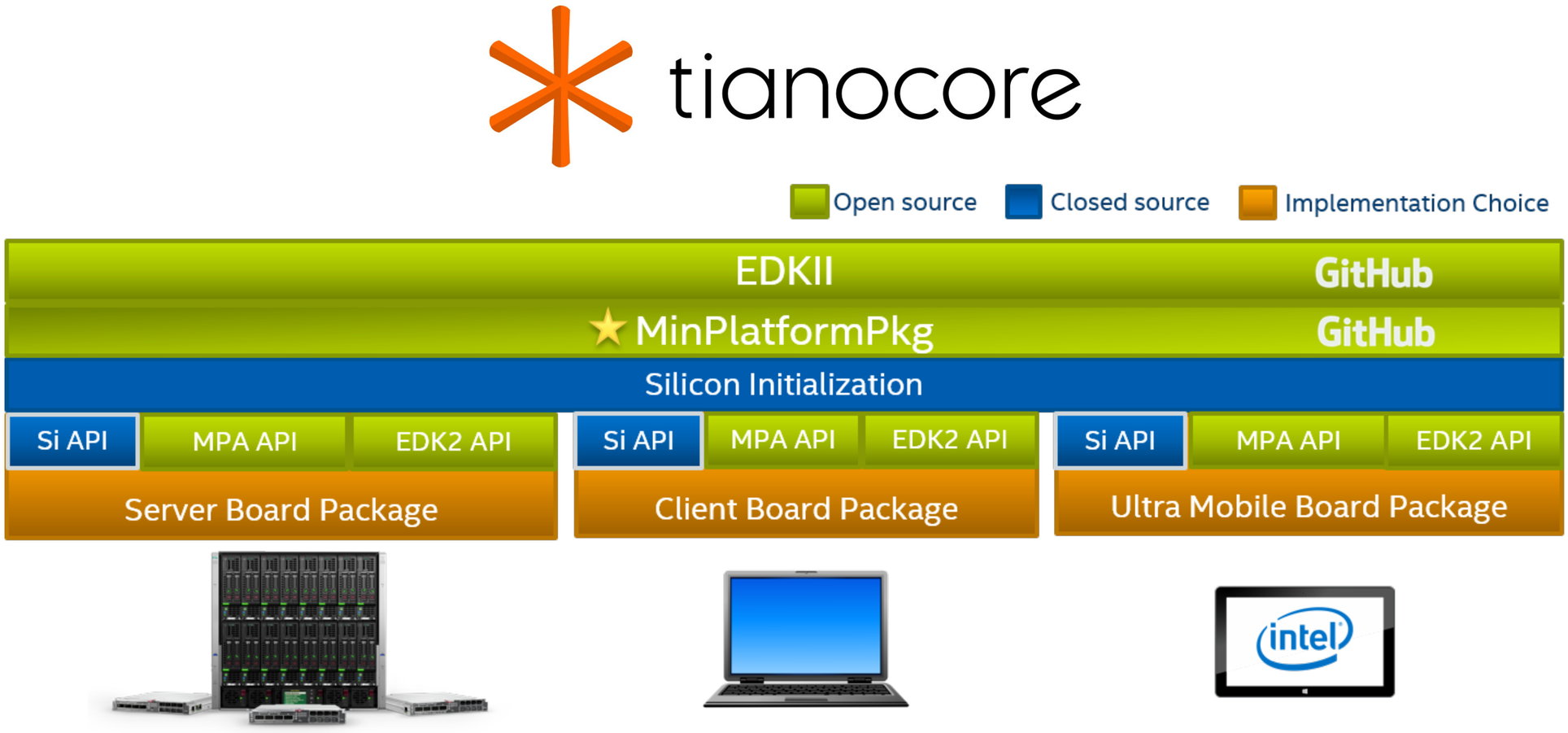
Open Firmware Using Linux

- One implementation of drivers
- Linux: Vetted for more than 20 years in military, consumer, and supercomputing systems
  - Already running on mission-critical devices around the world
  - Replace lightly-tested, closed drivers with hardened, heavily-tested, open source ones
- Bootstrap customization and fine-tuning for site-specific needs
  - More on the relevance to HPC upcoming
- More people understand Linux
  - Leverage existing talent/experience
- More mature tooling

Open Firmware at Facebook

- **2011:** Open Compute Project announced
- **2014:** OpenBMC: Open source baseboard management controller firmware
  - Now a Linux Foundation project
- **LISA18[4]/OSFC 2018[5]:** Facebook uses Linuxboot in the cloud
  - "Booting is hard"
    - Many different types of devices now vs. one *de facto* standard then
  - "More demands for firmware security"
    - Measured bootstrapping
  - "Provisioning is hard"
    - Firmware is now more complex
    - Need a robust provisioning solution

Intel



Source: https://software.intel.com/content/www/us/en/develop/articles/minimum-platform-architecture-open-source-uefi-firmware-for-intel-based-platforms.html

[6]

Tianocore EDKII

UEFI Firmware reference implementation

Overview

Arm is an active contributor to the EDKII project hosted by the Tianocore community.

UNIFIED EXTENSIBLE FIRMWARE INTERFACE

The EDKII project is an open source project that provides a modern, feature-rich, cross-platform firmware development environment for the UEFI and PI specifications developed and maintained by the UEFI Forum.

Arm contributions make sure the EDKII project constantly keeps an up to date implementation of a UEFI compliant firmware on Arm systems.

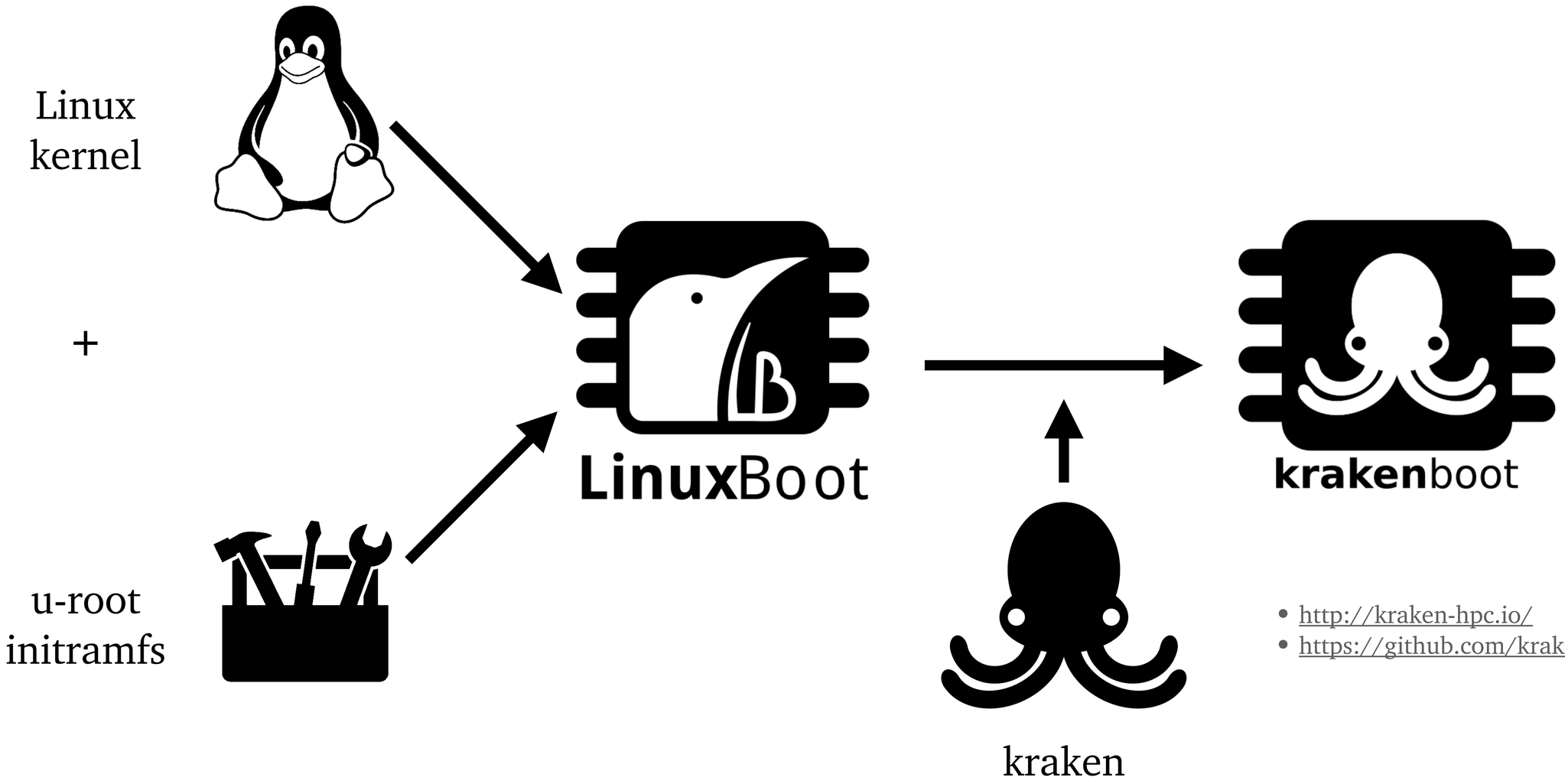
Arm contributes to both the EDKII main repository, maintaining some core packages like DynamicTablesPkg and StandaloneMMPkg, and the EDKII platforms repository, hosting support for various Arm reference platforms as well as other 3rd party Arm-based platforms maintained by either Linaro or partners.

Open Source Firmware in HPC

Motivation

- The need to boot systems most efficiently
  - Custom system initialization: greater control, finer performance
  - Boot times matter due to nodes more frequently rebooting
  - Vendor firmware is generic
- The need to use modern, secure protocols
  - TFTP, DHCP implementations can be buggy
  - HTTPS
  - TLS client certificates for cryptographic root-of-trust between nodes and parent
- The desire to run an extremely minimal operating system on compute nodes
  - Containerize user jobs directly on top of hardware
- The desire to potentially run a cluster state manager at a very low level to have better control of the nodes

Open Firmware at LANL



- http://kraken-hpc.io/
- https://github.com/kraken-hpc/

Sources and Further Reading

[1] Hudson, Trammell. “Bringing Linux back to the server BIOS with LinuxBoot”. *https://trmm.net/LinuxBoot_34c3/*. 2017-12-29.

[2] Ververis, Vassilios. “Security Evaluation of Intel's ActiveManagement Technology” *https://people.kth.se/~maguire/DEGREE-PROJECT-REPORTS/100402-Vassilios_Ververis-with-cover.pdf*.

[3] Minnich, Ron. “Replace Your Exploit-Ridden Firmware with Linux”. *https://youtu.be/iffTJ1vPCSo*. 2017-10-27.

[4] Hendricks, David; Barberio, Andrea. “Make Your System Firmware Faster, More Flexible and Reliable with LinuxBoot” *LISA18*. *https://www.usenix.org/conference/lisa18/presentation/barberio*. 2018-10-31.

[5] Hendricks, David; Barberio, Andrea. “Open Source Firmware @ Facebook”. *OSFC 2018*. *https://www.youtube.com/watch?v=eKVSBESoKUc*.

[6] Kubacki, Michael. “Minimum Platform: Open Source UEFI Firmware for Intel Based Platforms”. *OSFC 2019*. *https://www.youtube.com/watch?v=x3NFbUC3hkA*. 2019-12-02.

Questions?

Devon Bautista
dbautista@newmexicoconsortium.org

Lowell Wofford
lowell@lanl.gov

Join the Open Source Firmware Slack:
https://slack.osfw.dev/