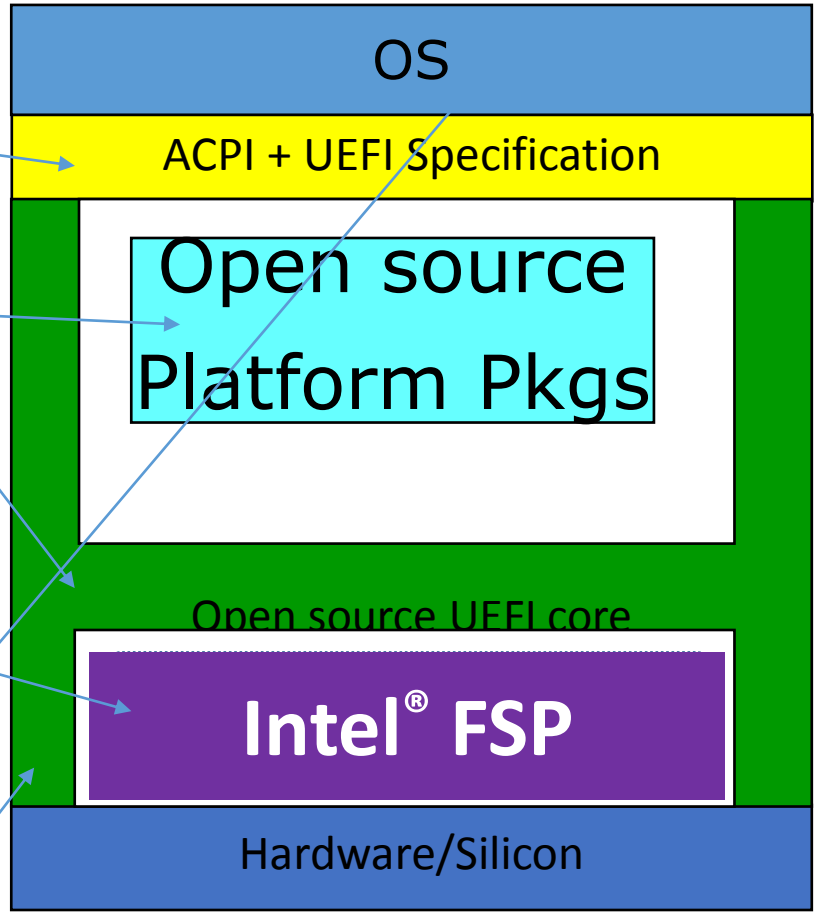
Potential OCP server approach

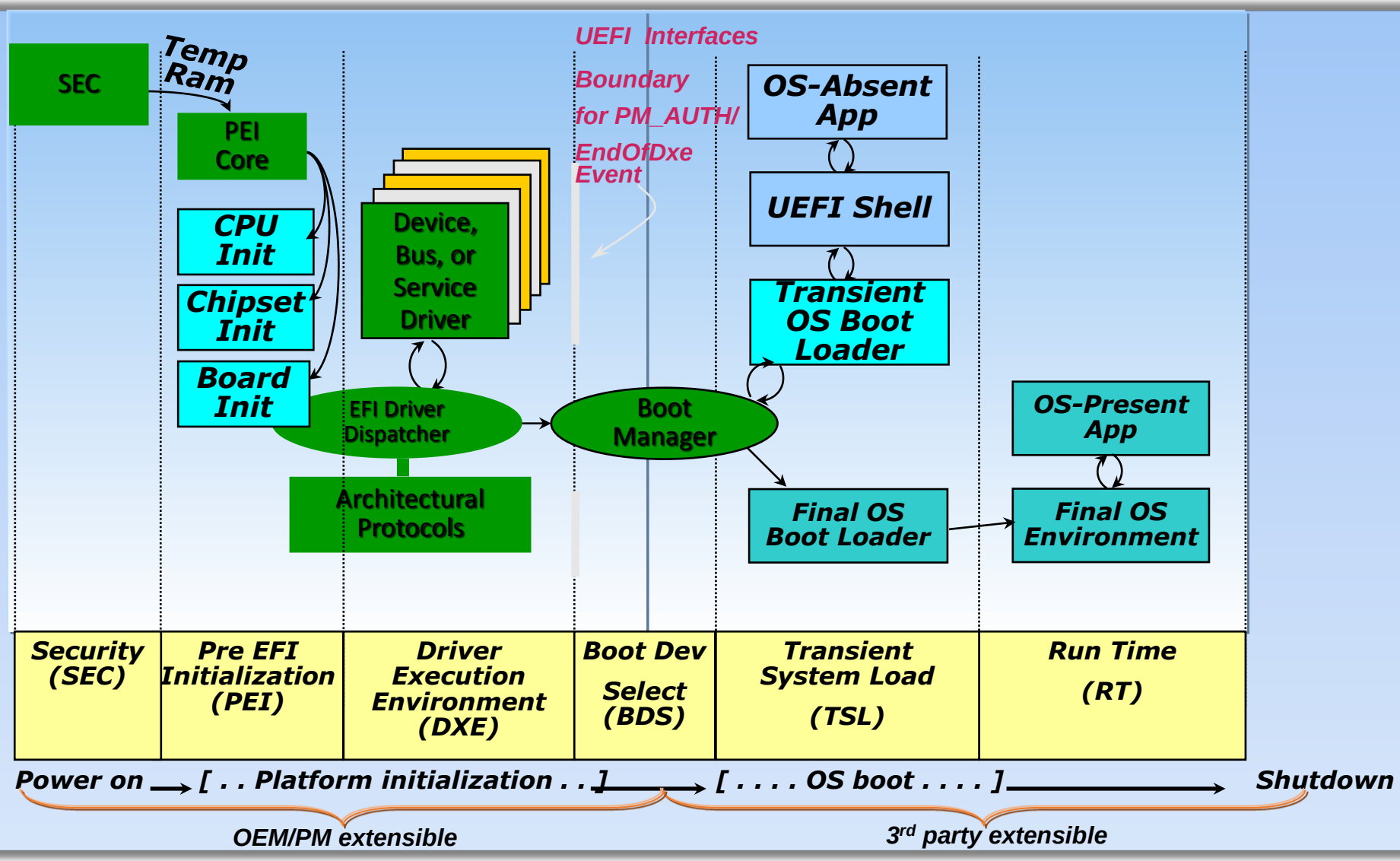
- Build entire platform from web content
- Boot open & shrinkwrap OS's via ACPI & UEFI spec www.uefi.org
  EDKII – existing upstream/open source core at https://github.com/tianocore/edk2
- And platform code w/ SMBIOS, ACPI + board https://github.com/tianocore/edk2-platforms
- Overview of design approach Min_platform
- Closed source SI Intel binary FSP for early IP-protected code https://github.com/intelfsp and/or other blobs https://github.com/tianocore/edk2-non-osi (e.g., binary PEI boot fw volume, other drivers)
- Can boot any UEFI OS from network, block media. Can embed OS (e.g., Linux) in flash for direct launch.
- Generic features – UEFI secure boot, TCG measured, pxe/http boot (including TLS), signed capsule update for host and device fw



UEFI PI based boot flow

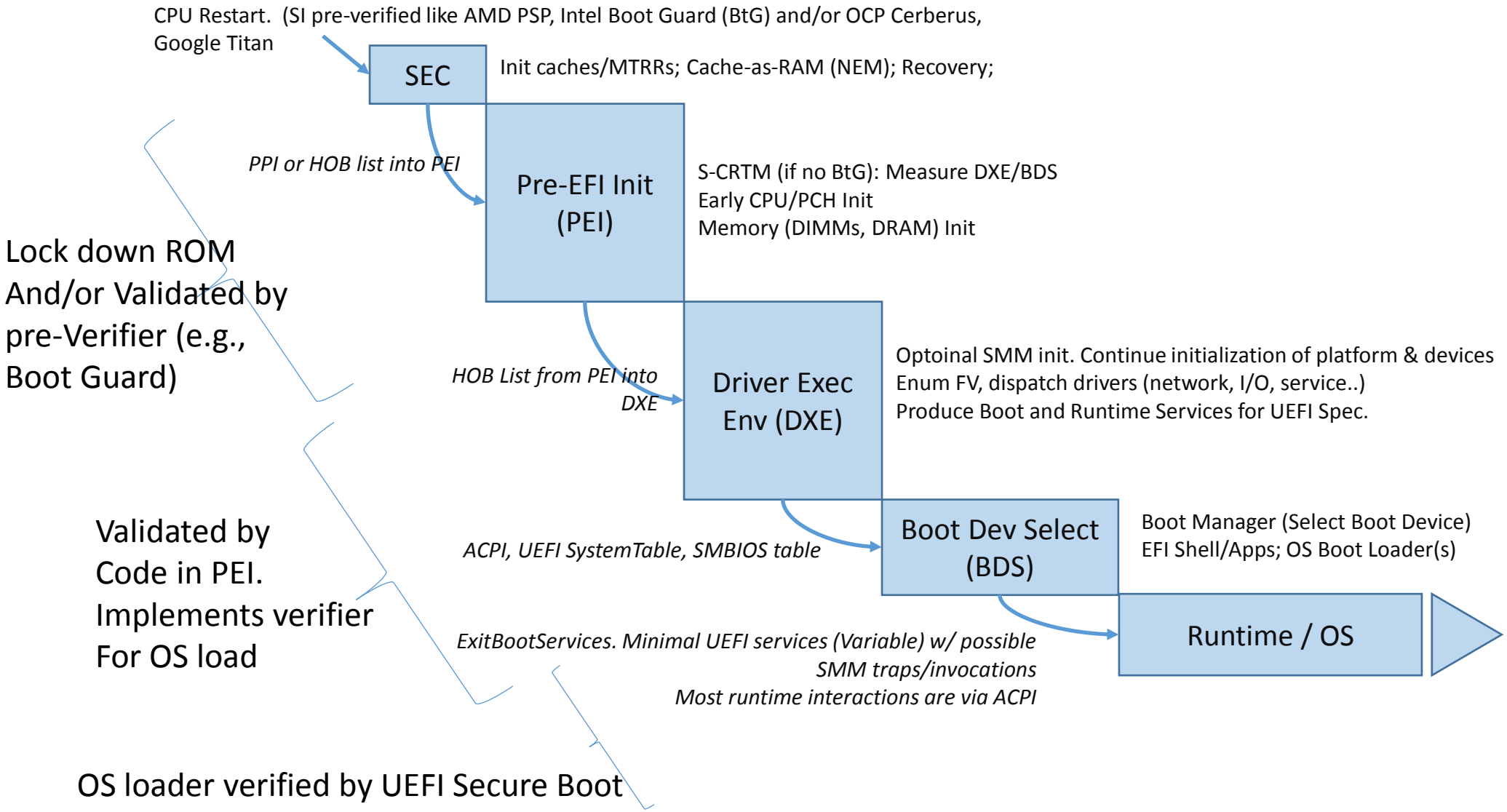
Non-host processor Verifier (Cerberus, Titan,..)

Service Processor/ BMC (Aspeed, other)

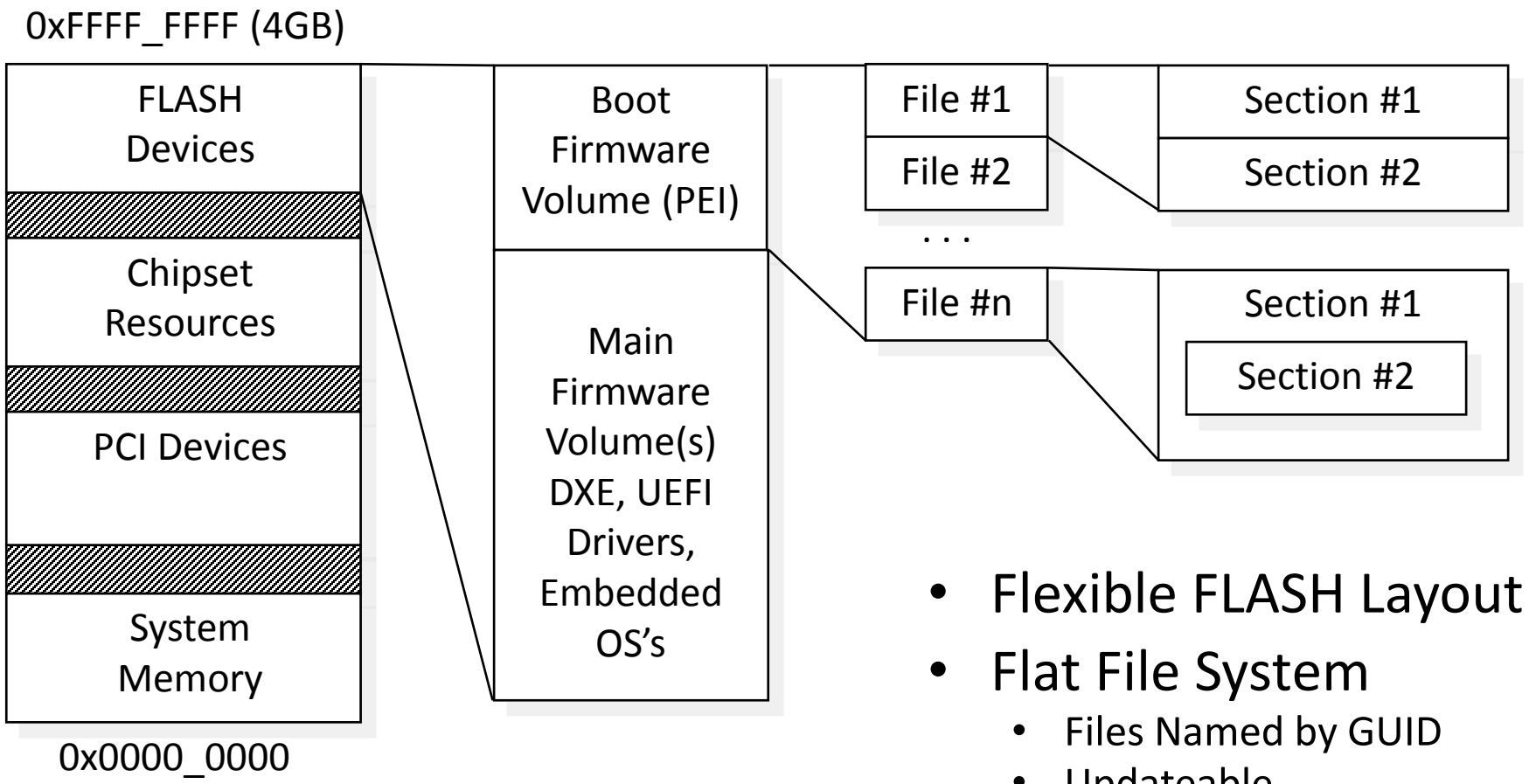


Code running on host CPU

UEFI / ACPI PI Firmware Flow



File System for FLASH Devices (FV's, FFS files)



- Flexible FLASH Layout
- Flat File System
  - Files Named by GUID
  - Updateable
- Extensible File Format

OS launch

- Can have an OS loader on network, disk, or in the SPI NOR flash
- EFI Device path points to where to find the OS loader
  - Can be a fixed device path or updatable via UEFI variable
- Linux can be launched as a single executable https://wiki.archlinux.org/index.php/EFISTUB
- UEFI Secure Boot allows for adding different OS loaders to a post ship system and maintain the chain of trust
  - Or can fix the certificates to lock down only a single OS target
- Firmware volumes (FVs) in flash can be partitioned at manufacture time to have minimum DXE to support – core, BDS, secure boot – in order to leave space for embedded OS target, or have alternate FV's with full feature UEFI drivers for devices in case of launching shrinkwrap OS's from various media

Approach – community discussion

- Have the binary blobs, recipes, and source code to build a full platform public
- Have the DXE FV's segregated so easy to have embedded OS or full UEFI support for shrink-wrap OS
- Use UEFI Secure boot and extensible key store to enable post-ship OS change

Challenges

- Extensibility in early part of system flow for post-ship devices
- Updates – OS specific, UEFI Capsule, Redfish,…