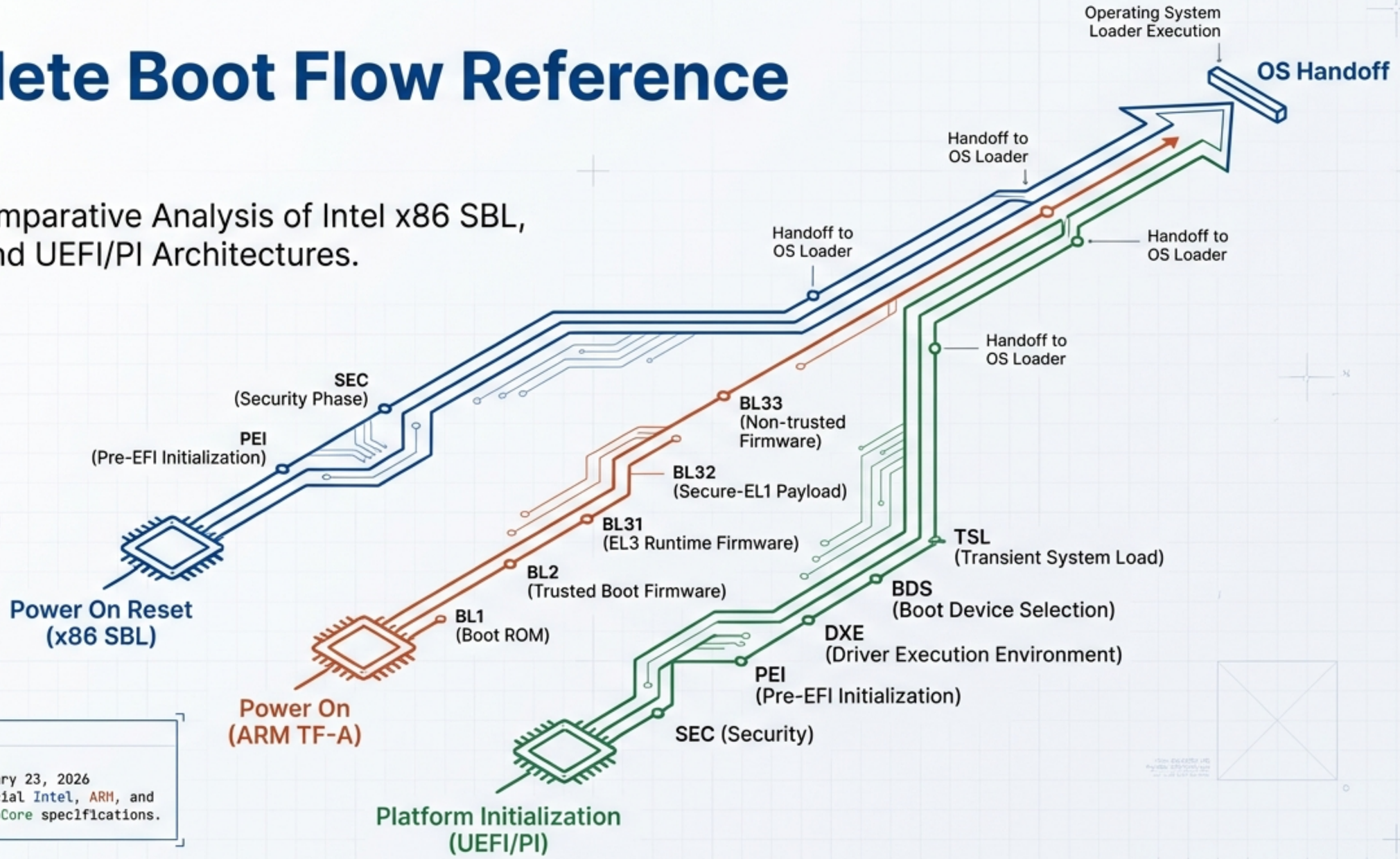
Inter Tight

Complete Boot Flow Reference

Inter

A Verified Comparative Analysis of Intel x86 SBL, ARM TF-A, and UEFI/PI Architectures.



JetBrains Mono

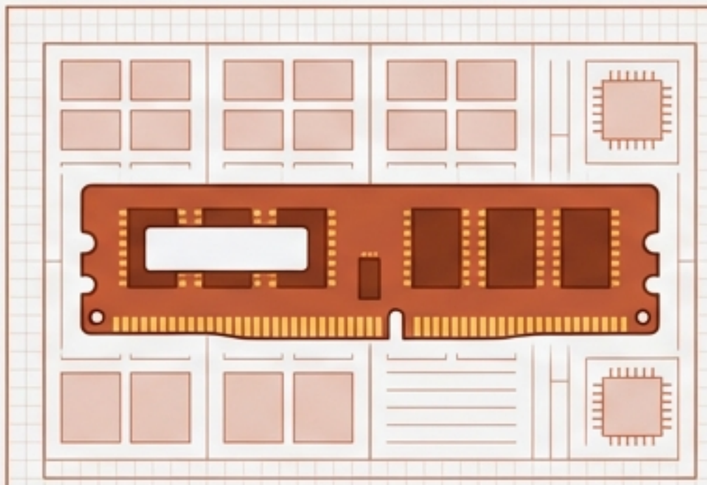
Document Version: 1.0
Verification Date: January 23, 2026
Verified Against: Official Intel, ARM, and TianoCore specifications.

The Universal Objective of Silicon Initialization

Regardless of the architecture, all modern boot flows typically address three non-negotiable requirements before handing control to the OS.

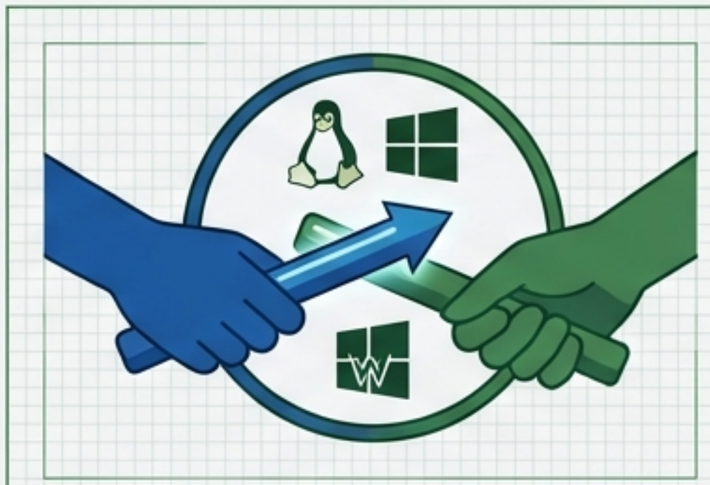
1. Hardware Root of Trust

Establishing integrity from the first instruction (Fuses/ROM).



2. Hardware Initialization

Configuring memory (DRAM) and silicon controllers.



3. OS Handoff

Securely passing hardware descriptions to the kernel.

Intel x86 (Arrow Lake)

Uses `Slim Bootloader (SBL)` with `Hardware-anchored chains.`

ARM (TF-A)

Uses `Exception Levels (EL)` and `Trusted Firmware-A.`

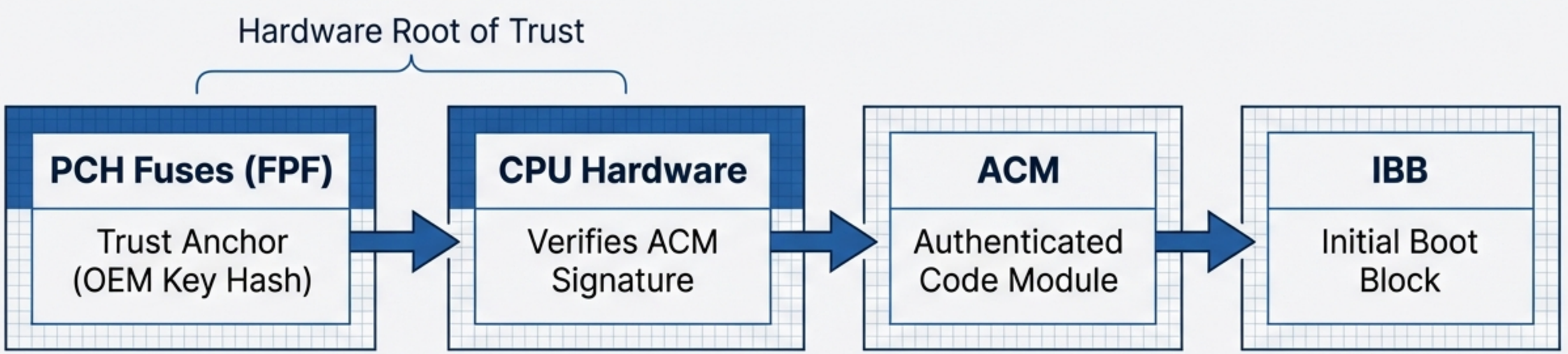
UEFI/PI

Uses a modular `Phase structure (SEC / PEI / DXE).`

Key Insight: While terminology differs (HOB vs. DTB, PEI vs. BL2), the security philosophy is convergent: a hardware-anchored chain of trust is mandatory.

Intel Arrow Lake Trust Model and Misconceptions

Trust Chain



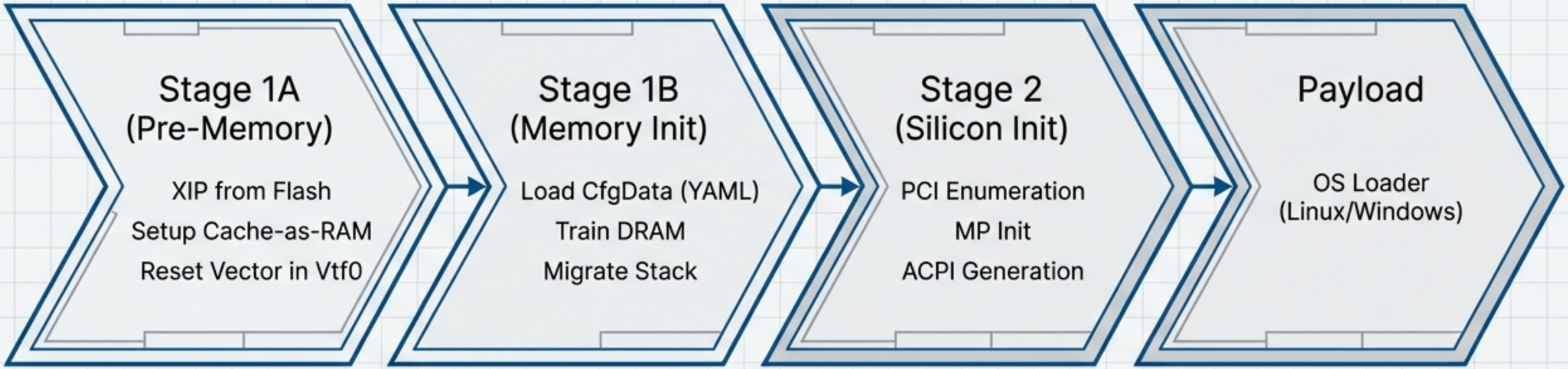
Clarifying CSME's Role

Myth: CSME acts as a runtime cryptographic co-verifier for the ACM.

Reality: The CSME enforces policies and authorizes manifests, but the CPU verifies the ACM signature using fused trust. The CSME is not a co-verifier in the runtime chain.

Reset Vector: 0xFFFFFFF0
Mechanism: Firmware Interface Table (FIT)

Intel SBL Execution and FSP Integration



Note: Creates 'LdrGlobal' state structure (internal use only).

FSP-T (TempRamInit)

Firmware Support Package - Temporary RAM Initialization

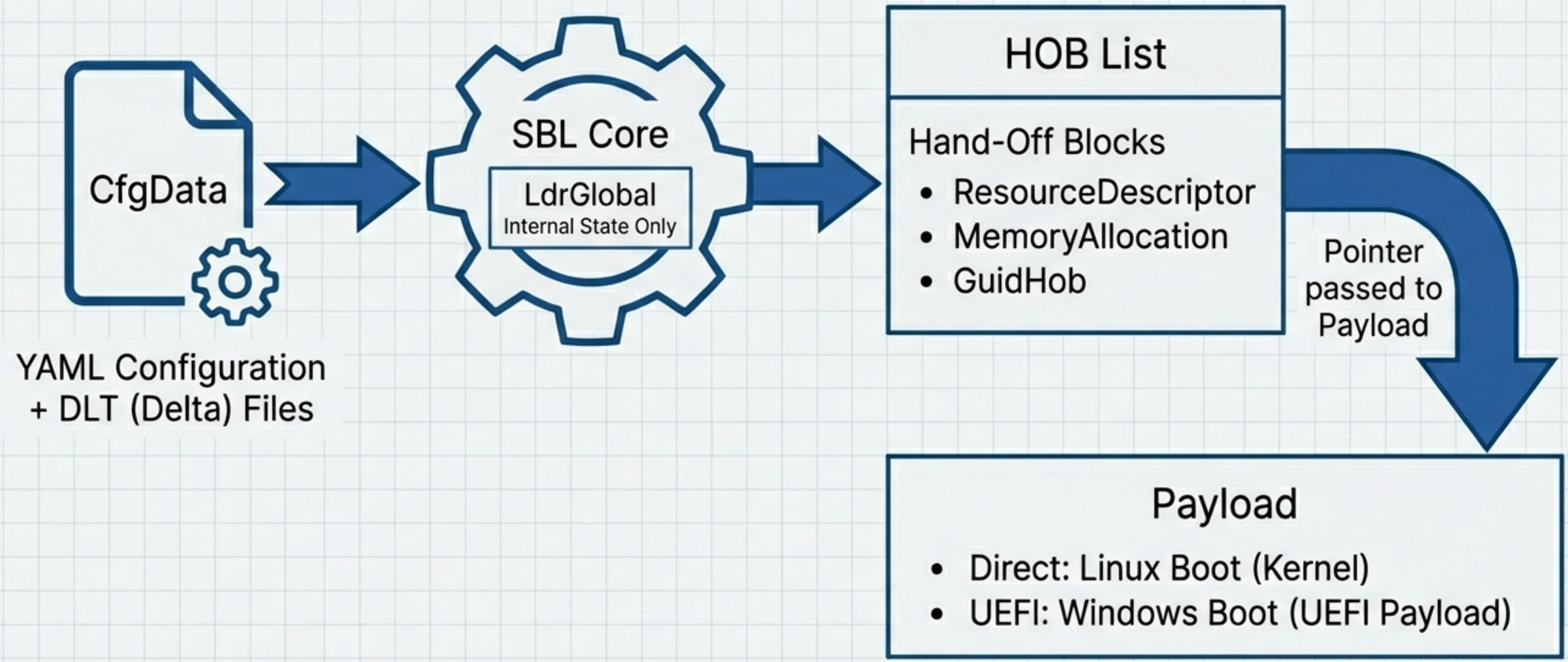
FSP-M (MemoryInit)

Firmware Support Package - Memory Initialization

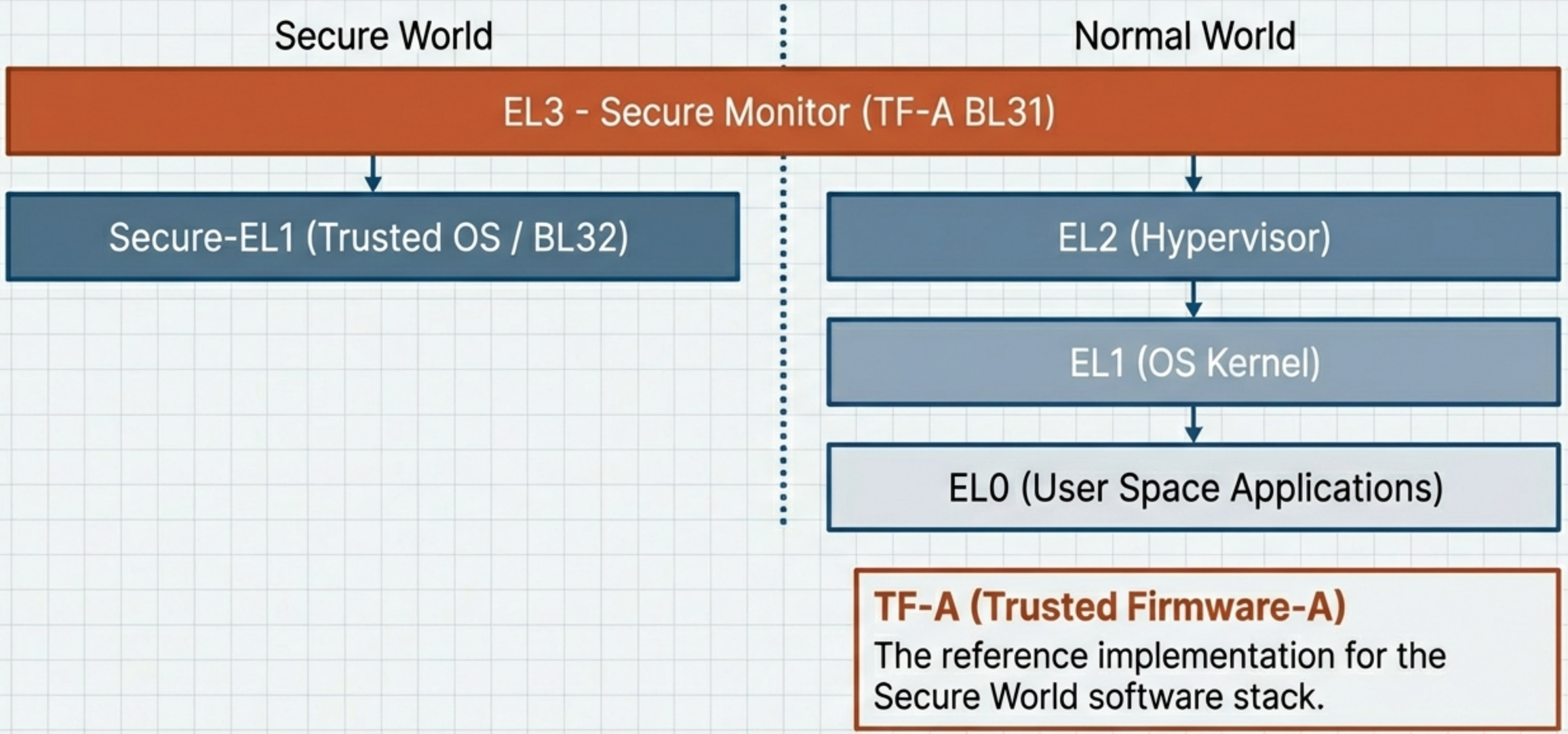
FSP-S (SiliconInit)

Firmware Support Package - SIlicon Initialization

Intel Data Structures and Handoff Mechanisms



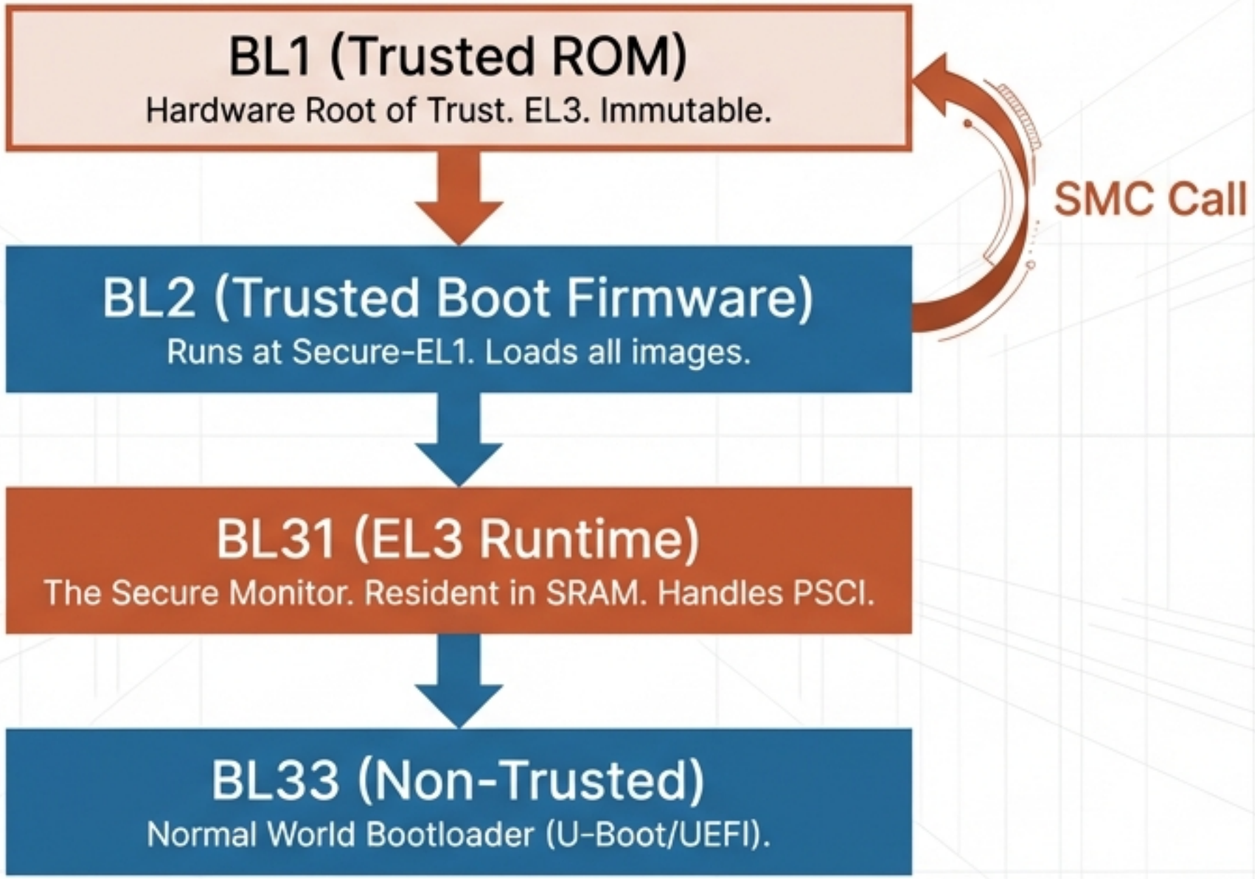
ARM Architecture: Exception Levels and Privilege



TF-A (Trusted Firmware-A)

The reference implementation for the Secure World software stack.

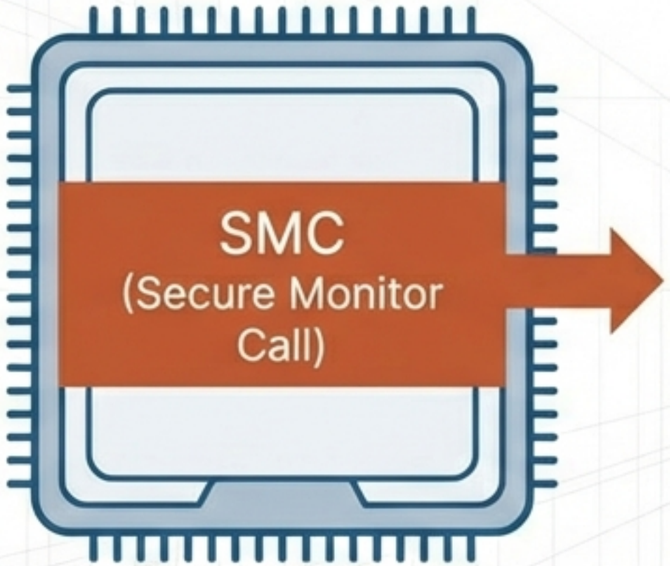
The ARM TF-A Chain of Trust (BL1 to BL33)



BL2 is the Loader, but BL31 is the **Runtime Manager** that facilitates the switch to the OS.

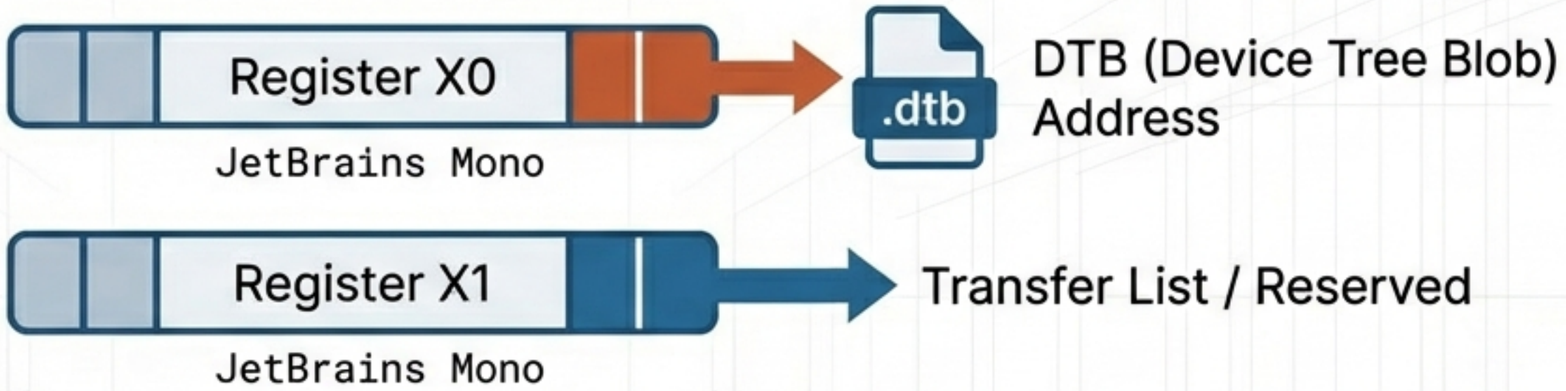
ARM Handoffs: SMCs and Device Trees

The Mechanism



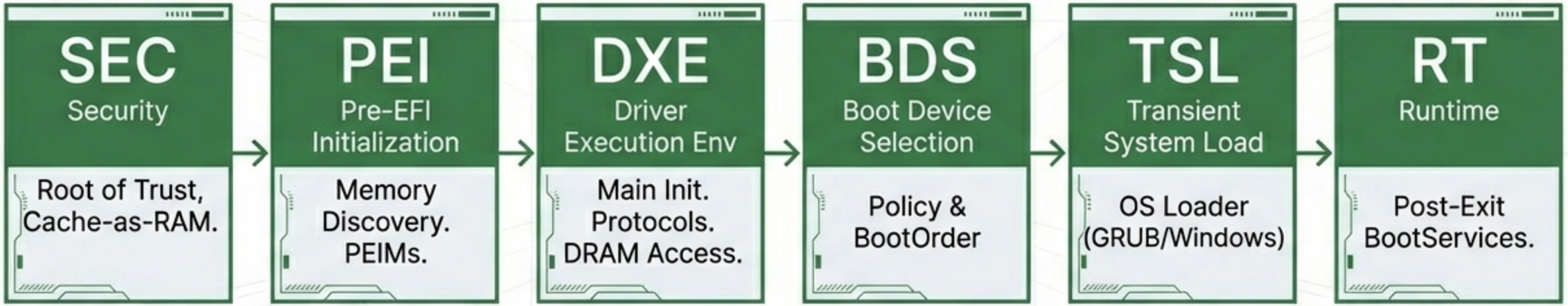
Used for transitions (BL2 -> BL1) and runtime services (OS -> BL31 for Power Management).

The Handoff



Hardware Description Format		
Embedded (Standard)	Device Tree (DTB)	Describes topology/interrupts.
Server (ServerReady)	ACPI	Generated by UEFI (BL33) for Enterprise OS.

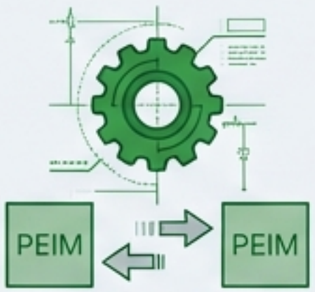
The Modular Structure of UEFI/PI

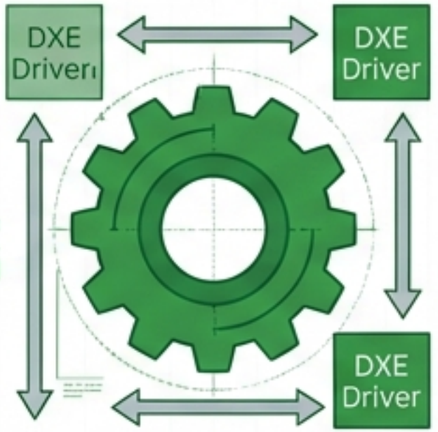


Bridging the UEFI Memory Boundary

PEI Phase (Pre-Memory)

DXE Phase (Main Memory)





PEIM
(XIP)

Comm Method:
PPIs

HOBs (Hand-Off Blocks)

- The only data survival mechanism across the memory boundary.
- Contains: Resource Descriptors, Memory Allocation, Firmware Volumes.

DXE Driver
(RAM)

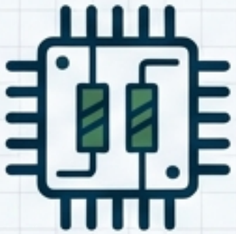
Comm Method:
Protocols

Architectural Phase Mapping (The Rosetta Stone)

	Intel x86 SBL	ARM TF-A	UEFI/PI
Root of Trust / Early Init	CSME Init / ACM	BL1 (Trusted ROM)	SEC Phase
Memory Initialization	Stage 1B + FSP-M	BL2	PEI Phase
Silicon Init	Stage 2 + FSP-S	BL2 / BL31	DXE Phase
Runtime / OS Load	Payload / SMM	BL31 / BL33	BDS / Runtime Services

Comparative Trust Anchors and Security

Intel x86 (CBnT)



Anchor: PCH Field Programmable Fuses (FPF)

CPU verifies ACM ACM verifies IBB

CPU → ACM → IBB

ARM TF-A

Anchor: Trusted ROM (Immutable Code)

BL1 verifies BL2 BL2 verifies BL31

BL1 → BL2 → BL31

UEFI Secure Boot

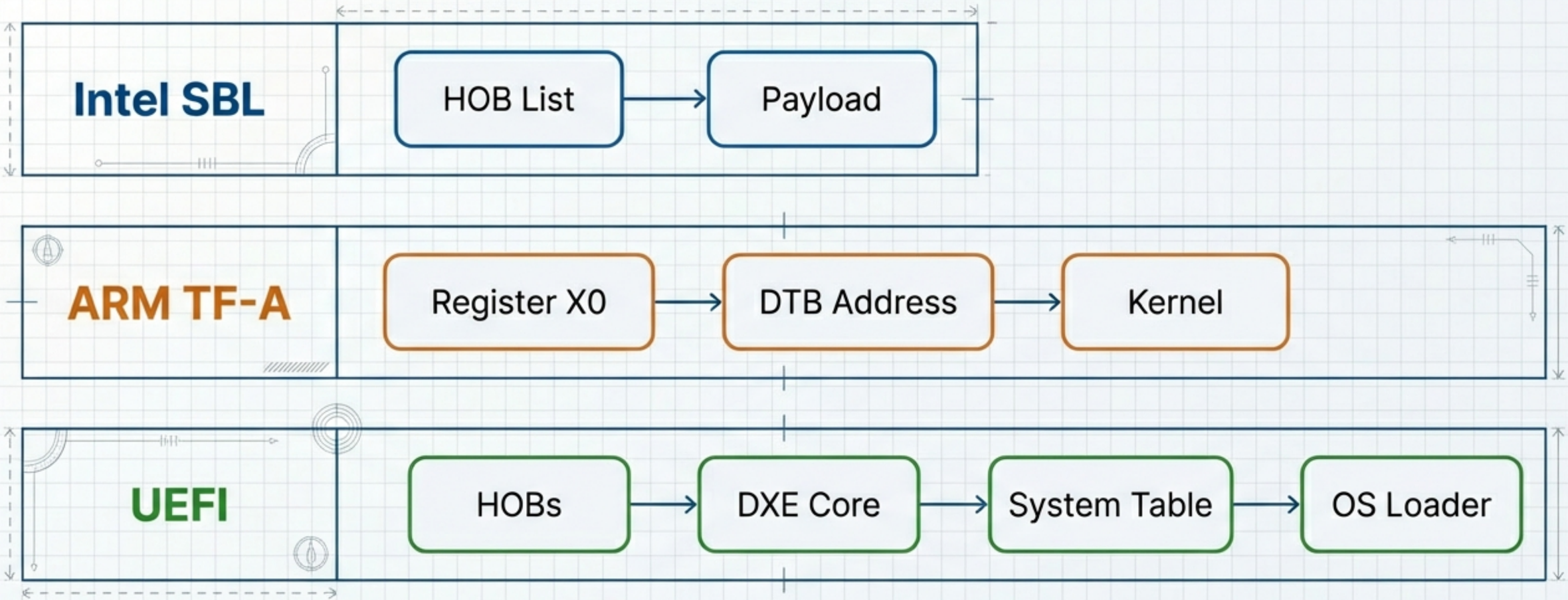
Anchor: Platform Key (PK) in NVRAM

Signature verification against db/dbx databases

Signature Verification → db/dbx databases

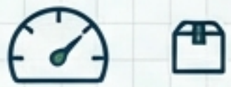
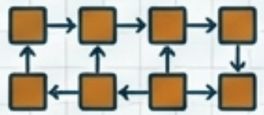
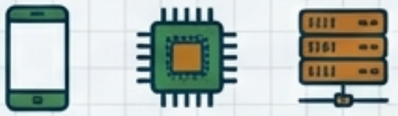
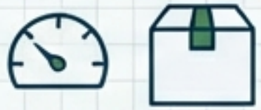
Intel and ARM rely on immutable hardware/fuses for the initial anchor; UEFI Secure Boot typically relies on **NVRAM** storage verified by a prior stage.

Data Handoff Mechanisms to the OS



Key Distinction: Intel SBL and UEFI rely heavily on HOBs for internal passing; ARM relies on the Device Tree (DTB) passed via register convention.

Ecosystem Landscape and Use Cases

	Speed/Size	Flexibility	Primary Target
Intel SBL	Fast / Small 	Limited (YAML) 	Embedded x86, Client, IoT 
ARM TF-A	Fast / Small 	High (Modular Stages) 	Mobile, Embedded, Server 
UEFI (EDK II)	Slower / Large 	Very High (Drivers) 	General Purpose Desktop/Server 

Takeaway: SBL and TF-A optimize for speed and specific silicon;
UEFI optimizes for broad compatibility and plugin-style driver support.

Glossary and Official Verification

ACRONYMS

ACM: Authenticated Code Module

BL: Boot Level (ARM)

CSME: Converged Security and Management Engine

HOB: Hand-Off Block

SMC: Secure Monitor Call

XIP: eXecute In Place

VERIFICATION SOURCES

Intel Boot Guard / CBnT Arrow Lake Datasheets (2026)

Slim Bootloader GitHub Documentation

ARM Trusted Firmware-A Design Documents

UEFI Specification v2.10+ / TianoCore EDK II