ARM Trusted Firmware

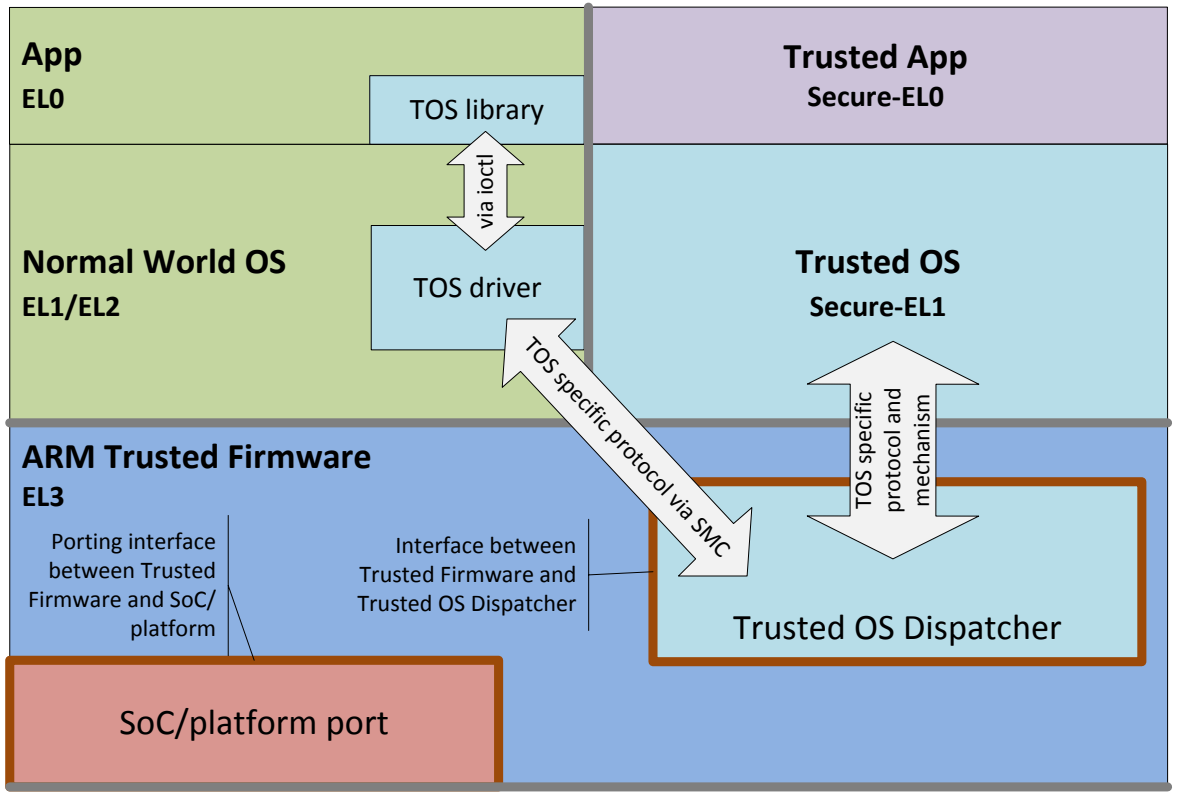
LCA14 – March 2014

Andrew Thoelke
Systems & Software, ARM

ARM Trusted Firmware for 64-bit ARMv8-A

A refresher

- Standardized EL3 Runtime Firmware
  - For all 64-bit ARMv8-A systems
- Reducing porting and integration work
  - For SoC and Trusted OS developers
- Reusable, reference implementations
  - PSCI
  - SMC Calling Convention
  - Configuration of ARM hardware
- Running on ARMv8-A FVPs and Juno
  - … and on partner's silicon



SoC supplier	ARM Trusted Firmware	Trusted App supplier
OS/hypervisor supplier	Trusted OS supplier	Internal TOS interface

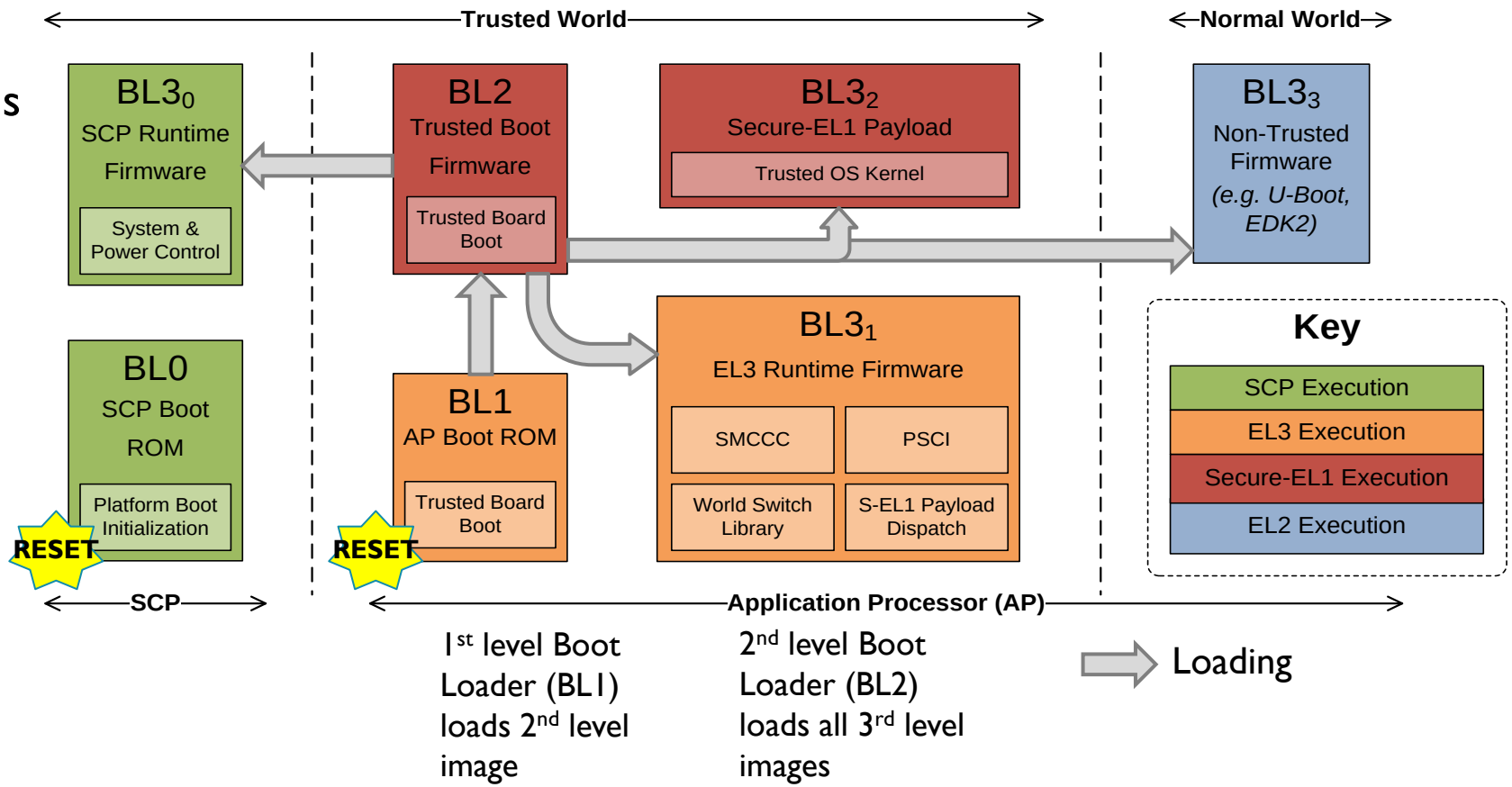
https://www.github.com/ARM-Software/arm-trusted-firmware

ARM

ARM Trusted Firmware for 64-bit ARMv8-A

A refresher

- Reference boot flows
  - For 64-bit ARMv8-A systems
- Open Source at GitHub
  - BSD License
  - Contributors welcome
- We just released v1.0
  - A reason to celebrate?



https://www.github.com/ARM-Software/arm-trusted-firmware

ARM

What's happened since last time?

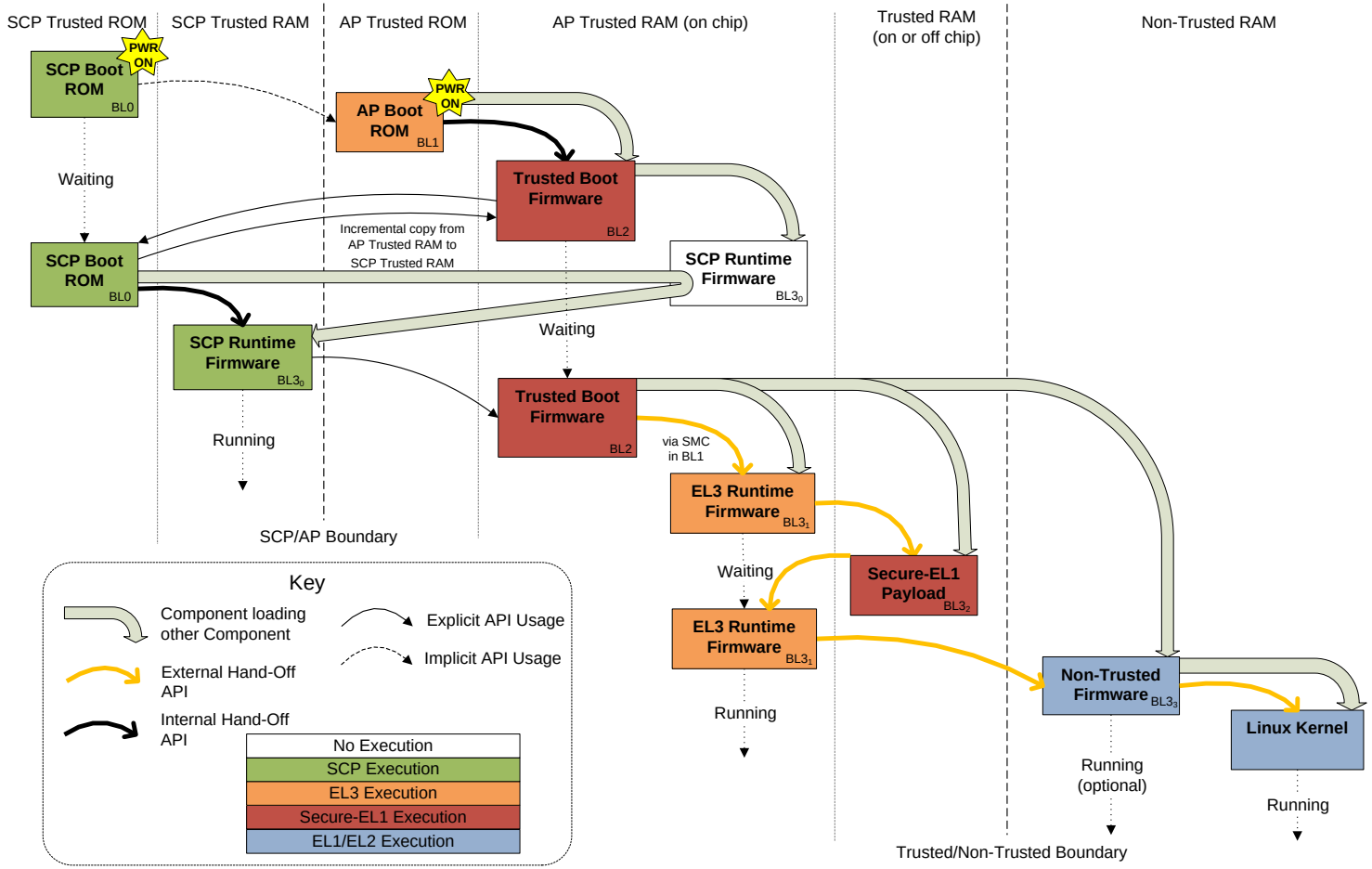
https://www.github.com/ARM-Software/arm-trusted-firmware

ARM

Juno

ARM Trusted Firmware on ARMv8-A silicon

- Juno port upstream
  - Complete PSCI implementation
  - Application processor firmware is all open for updating by developers
- Still to come:
  - More secure RAM for Trusted OS porting and development
  - Authentication of firmware images during boot



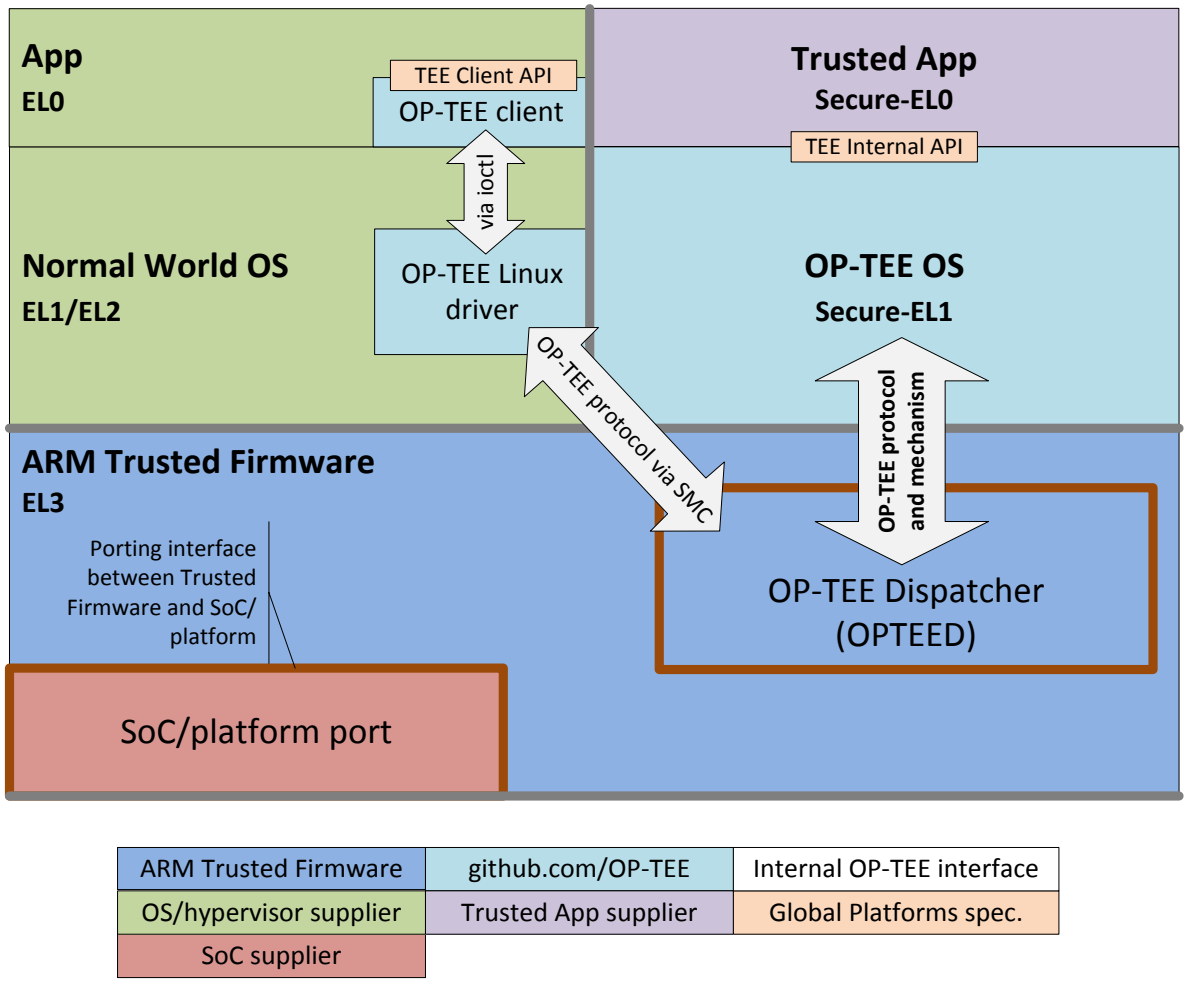
Detailed boot flow on Juno

https://www.github.com/ARM-Software/arm-trusted-firmware

Trusted OS and TEE

Support for 3rd party Trusted OS/TEE

- Support for secure interrupts and secure DDR RAM
  - Supporting different Trusted OS memory and interrupt requirements
- OP-TEED patches merged this week
  - Thank you Jens!
  - OP-TEE now running on ARMv8-A FVP models – all code upstream
- Still wanted:
  - OP-TEE running on ARMv8-A silicon
  - Dispatchers for other OSS Trusted OSes



https://www.github.com/ARM-Software/arm-trusted-firmware

ARM

Production Platforms

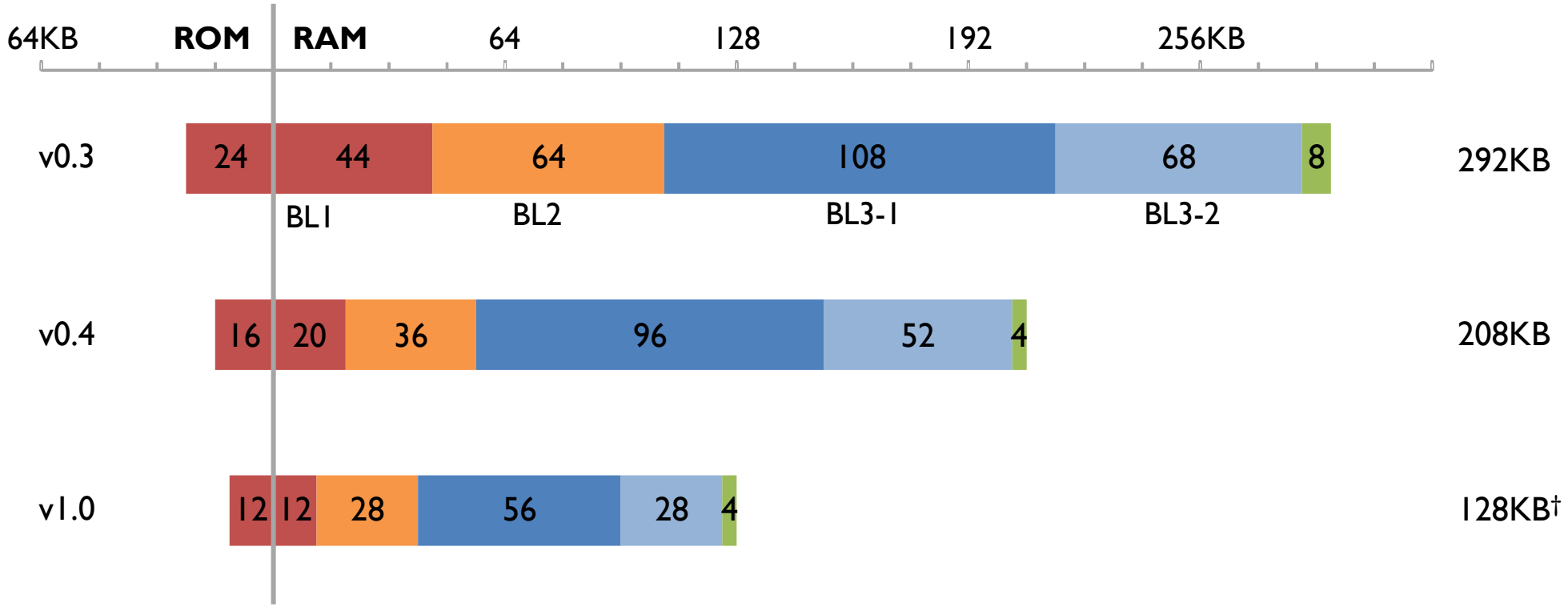
ARM Trusted Firmware running on partners' ARMv8-A silicon

- Alternative boot flows supported
  - Reuse of existing secure boot loaders
  - Reset to RAM firmware
  - Stable boot flow interfaces
- Improved debugging of firmware errors
  - Crash reporting for fatal errors and unexpected exceptions
- Easy selection of 32/64-bit execution
  - Register width for Trusted OS
  - Exception Level and register width for normal world software
- Platforms with wide I/O addresses
- "Follow the manual" CPU specific code
  - For correct hardware operation
- Still to come
  - Performance and scalability investigation and improvements
  - Even more boot flow flexibility – Enterprise and Networking scenarios
  - Improving integration for SoC suppliers, Trusted OS suppliers and OEMs

https://www.github.com/ARM-Software/arm-trusted-firmware

Shrinking firmware

The evolution of ARM Trusted Firmware's memory footprint*



* Memory usage of code and data for a release build of ARM Trusted Firmware for FVP including the TSP
† In v1.0 the peak memory usage is even lower as some of the firmware images are overlayed in the same memory during the course of booting the platform

https://www.github.com/ARM-Software/arm-trusted-firmware

ARM

Next time in ARM Trusted Firmware…

- ARM is planning to have contributed
  - A reference implementation of Trusted Board Boot, up to the non-secure firmware
  - Support for PSCI v1.0 – a specification update is in progress

- What else shows up depends on you
  - Tell us what's broken
  - Tell us what's missing
  - Send us your improvements … new contributions are always welcome

https://www.github.com/ARM-Software/arm-trusted-firmware

Thank you

https://www.github.com/ARM-Software/arm-trusted-firmware

ARM