Arm Server Ready

Dong Wei
Agenda

● Arm ServerReady Program
● SBSA/SBBR Updates
● PCIe Integration Updates
● UEFI Forum Updates
● Server Management Strategy
Agenda

- Arm ServerReady Program
- SBSA/SBBR Updates
- PCIe Integration Updates
- UEFI Forum Updates
- Server Management Strategy
Vision

○ Arm servers everywhere and easy to deploy

Mission

○ We provide the tools to enable customers to deploy Arm servers with confidence

Elevator

○ ServerReady gives the confidence that your server works out of the box.
Arm has been investing in the creation of a certification program. The Arm Server Ready Program will consist of:

- A set of test suites covering our standard specs, and additional OS image boot, install, and network tests.
- On the ground support for ODMs, where the rubber hits the road.
- Logo and marketing materials to be used on certified platforms.
Platform Architecture

Base System Architecture (BSA)
- Defines hardware requirements

Base Boot Requirements (BBR)
- Defines firmware requirements

These specifications require a minimum set of hardware and firmware implementations that will ensure OS and firmware will interoperate

SBSA/SBBR are the BSA/BBR for the server systems
- Developed using feedback from vendors across the industry (Silicon vendors, OSVs, Hypervisor vendors, BIOS vendors, OEMs and ODMs)
- SBBR defines the required, recommended and optional UEFI, ACPI and SMBIOS interfaces

SBSA are SBBR are now available at https://developer.arm.com/
- Current versions are SBSA v3.1 and SBBR v1.0. No click through license required.
SBBA and SBBR Architectural Compliance Suites

SBBA test covers
- SBSA CPU properties
- SBSA defined system components
- SBSA rules for PCIe integration
  - Based on the PCIe specification
  - Based on standard OS drivers with no quirks enabled

SBBR test covers
- UEFI testing based on the UEFI SCT
- ACPI testing based on FWTS
- SMBIOS testing

V1.0 released!
- [https://github.com/ARM-software/sbsa-acs](https://github.com/ARM-software/sbsa-acs)
- [https://github.com/ARM-software/arm-enterprise-acs](https://github.com/ARM-software/arm-enterprise-acs)
Agenda

- Arm ServerReady Program
- SBSA/SBBR Updates
- PCIe Integration Updates
- UEFI Forum Updates
- Server Management Strategy
Drafting SBSA v5.0

Some highlights:
- Single doc with L3/L4/L5
- PCIe updates: no OS observable PCIe enhanced allocation is permitted
- Fast counter support
- Closer alignment with 8.2 in Level 4 and 8.4 in Level 5
- Under investigation
  - TCG TPM
  - Longer term we want to ban non-standard interrupt controllers but only after we have provided some suitable alternatives
Drafting SBBR 1.1
Requires newer FW revisions
  • ACPI6.2, UEFI2.7, SMBIOS 3.1.1, PSCI 1.1

Require PSCI as the only secondary core boot method

Require AArch64 native UEFI Drivers and Applications

Newer features
  • Generic Event Devices and interrupt-signalled Events
  • Software Delegated Exception
  • Heterogeneous Memory Attribute
  • Redfish Host Interface
  • Under Investigation
    ■ TCG TPM Trusted Boot
    ■ UEFI Secure Boot
    ■ Arm TF secure boot
Server Standards Roadmap

- **Specs**
  - SBSA-5.0 – Alpha 2
  - SBSA/SBBR v1.0
  - SBSA/SBBR Compliance process
  - SBSA-5 (including L4)
  - SBSA/SBBR v2.0
  - SBSA/SBBR next

- **SBBA/BR Test suites**
  - SBSA– beta * PCIe testing alpha
  - SBSA/SBBR v1.0
  - SBSA/SBBR Compliance process
  - SBSA-5 (including L4)
  - SBSA/SBBR v2.0
  - SBSA/SBBR next

- **Future**
  - SBSA/SBBR next

**Timeline**
- C2017 Q2
- C2017 Q3
- C2017 Q4
- C2018 Q1
- C2018 Q2
- Future

- Released
- Development
- Adv. Planning
- Concept
- Ongoing updates
Agenda

- Arm ServerReady Program
- SBSA/SBBR Updates
- PCIe Integration Updates
- UEFI Forum Updates
- Server Management Strategy
ARM’s plan to ensure “works out of the box” PCIe

- Enhance SBSA to ensure platforms have a PCIe feature set that addresses infrastructure market needs
- Enhance SBSA to ensure that rules are rigorously specified where multiplicity of options/ambiguity exists
- Add corresponding compliance tests
- Enable/Create a “SBSA compliance verification” endpoint HW/Transactor to get exhaustive coverage
ARM’s plans for enabling creation of “performant” PCIe subsystems

- Create a guide for usage model driven PCIe performance verification
- Enable/create a “performance verification” endpoint HW/Transactor
- Create a “best practices” guide for building functionally correct PCIe I/F with high performance
Agenda

- Arm ServerReady Program
- SBSA/SBBR Updates
- PCIe Integration Updates
- UEFI Forum Updates
- Server Management Strategy
UEFI Forum

Arm Limited is now a Promoter of the UEFI Forum!
- Board member
- Vice President (Chief Executive)
- Chair of the UEFI Test WG
- Co-Chair of the ACPI WG
ACPI

ACPI 6.2 was released in May
  ○ Better support for cache topology discovery
  ○ Improved PCC channels
  ○ Alignment with Software Delegated Exceptions
  ○ IORT updated also in May which improved SMMUv3 support

Current work
  ○ PCC operating regions: better ways for ASL to talk to platform controllers
  ○ CoreSight
  ○ SMMU and RAS
  ○ MPAM

Anything else we should be looking at?
UEFI

UEFI 2.7 was released in May, no major updates affecting Arm bindings.

Looking forward 8.2 support for 52 bit PA and VA is our biggest problem:
- Requires 64KB page mappings where as UEFI is built on a strong 4KB mandate
- Affects spec as well as code

SCT
- UEFI v2.6 SCT is accelerated (Final Draft Candidate ready in July pending Board approval)
- Investigating new development model
Agenda

- Arm ServerReady Program
- SBSA/SBBR Updates
- PCIe Integration Updates
- UEFI Forum Updates
- Server Management Strategy
Server Management – Problem Statement

OS/Hypervisor

- Common OS-SoC interface
- Common OS-BMC interface

ODMs

- Common OS-BMC interface
- Common BMC-SoC interface

System Admins

- Redfish API

---

**Diagram:***

- Host (UEFI, OS)
- SoC Elements (e.g. sensors)
- On-chip Management
- BMC
- Platform Elements
- System Admin
Arm Proposal

Joined DMTF/SPMF

Create a Server Base Management Guide (SBMG) specification
  ○ Level 0: The current designs by SiPs
  ○ Level 1: Standardize with the currently mature technologies
  ○ Level 2: Standardize with more capable technologies

Participate in the UEFI/DMTF collaboration on creating a solution for BIOS configuration remote deployment
Level 1: Standardize on SCMI/PCC, Redfish, IPMB/I2C and PMCI(MCTP/PLDM)

Host (UEFI, OS)

SCMI or ACPI/PCC

SoC Elements (e.g. sensors)

On-chip Management

Redfish Host Interface

In-band

BMI

Out-of-band

SoC

Platform Elements

SCMI or ACPI/PCC

PCIe NIC

IPMB/I2C

PLDM/MCTP

MCTP compliant Transport: PCIe, or I2C/SMBus (SMLink)

Arm/ACPI compliant transport

Redfish API
Level 2:
Standardize on SCMI/PCC, Redfish and PMCI(MCTP/PLDM)

- Host (UEFI, OS)
- SoC Elements (e.g. sensors)
- On-chip Management
- Platform Elements
- BMC

---

SCMI or ACPI
PCC

SoC

Redfish Host Interface

Platform Elements

MCTP compliant Transport: PCIe, or I2C/SMBus (SMLink)

Arm/ACPI compliant transport
Work To Do

SCMI/PCC Standardization

Redfish Enablement on UEFI
- Arm to participate in UCST and Tianocore work to make sure it works with Arm

BMC-less Strategy
- Arm architecture (not vendor specific) can try to define another message-base interface

OpenBMC
- [https://github.com/openbmc/openbmc](https://github.com/openbmc/openbmc)
- Mega Data Centers requests OpenBMC solution
- Arm community (LEG?) should provide a maintainer for the OpenBMC project to make sure OpenBMC works well for Arm servers
- Redfish implementations for OpenBMC (already done, but not upstreamed?)
Embedded Base Boot Requirements

SFO17-508 11:30am Friday
Please Join if Interested
Thank You

#SFO17
SFO17 keynotes and videos on: connect.linaro.org
For further information: www.linaro.org