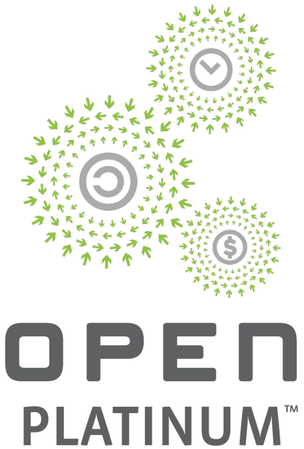
Open. Together.

OCP
SUMMIT

OPEN SYSTEMS FIRMWARE

Open Firmware DRAM Training Data Interface

Natarajan Sivagar, Software Engineer, Google Inc



Agenda

OPEN SYSTEMS FIRMWARE

- Advantages of Open Source and Standards
- Filling Gaps in Standards
- Understanding Server DRAM Software Stack
- Firmware Memory Data Interfaces
- Proposed Data Interface
- Call for Action

Embedded Software

Advantages of Open Source and Standards

OPEN SYSTEMS FIRMWARE

- Open Source projects yields quality software with robust security features
- Provides optionality in terms of options and features
- We have seen benefits in our adoption of
  - Linux
  - OpenBMC and
  - (now in) LinuxBoot
- We are committed to contribute and create open standards

Industry Standards and Benefits

OPEN SYSTEMS FIRMWARE

- For firmware there are various Industry Standards for interfaces like DMTF specifications, ACPI, etc.
- These standards provide
  - Ability to create standard drivers which enhances interoperability
  - Ability to introduce newer systems quicker
    - Instead of writing a new driver from scratch we can utilize existing driver for early testing/development

Background: DIMM Training

OPEN SYSTEMS FIRMWARE

“Training" is the process of iteratively putting data on a bus, observing bus behavior, and tuning *per-wire* on-chip delay lines to optimize performance and minimize error.”

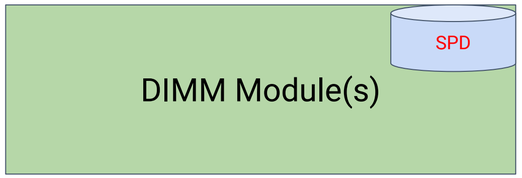
- Ron Minnich (Google)

(http://tinyurl.com/cisl2012-coreboot)

DIMM Software Stack: SPD

OPEN SYSTEMS FIRMWARE

- DIMM Modules contains Serial Presence Detect (SPD) data in a non-volatile storage
- SPD data is accessed through SMBus
- SPD provides relevant info about the DIMM like the DRAM organization, supported timings, serial & product name/number, manufacturer, etc.
- Contains other proprietary maintenance data such as Post Package Repair info, fused cell info etc.

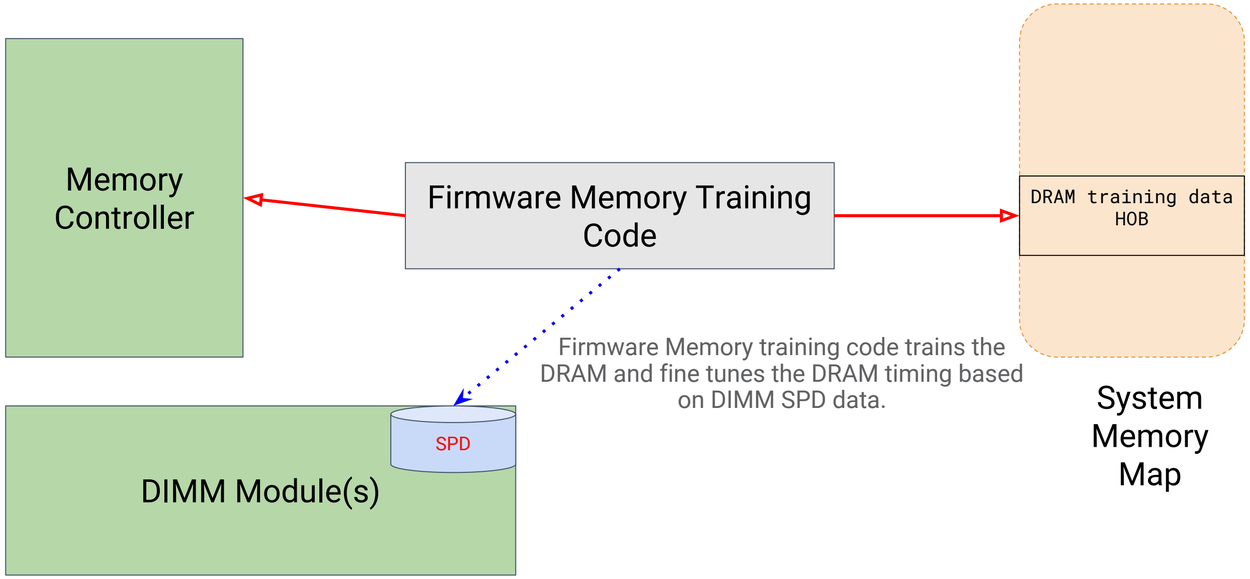


- Data in SPD is useful during DIMM repair flow
- Reading thru SMBus during runtime is not always feasible
- BMC based servers uses out of band mechanisms like PECI to access SPD

DIMM Software Stack: Training

OPEN SYSTEMS FIRMWARE

- Memory reference code trains the DRAM timings based on SPD data, OEM/Board configuration and marginining algorithms
- MRC programs the memory controller with right timing values
- MRC disables DIMM/memory channels as required

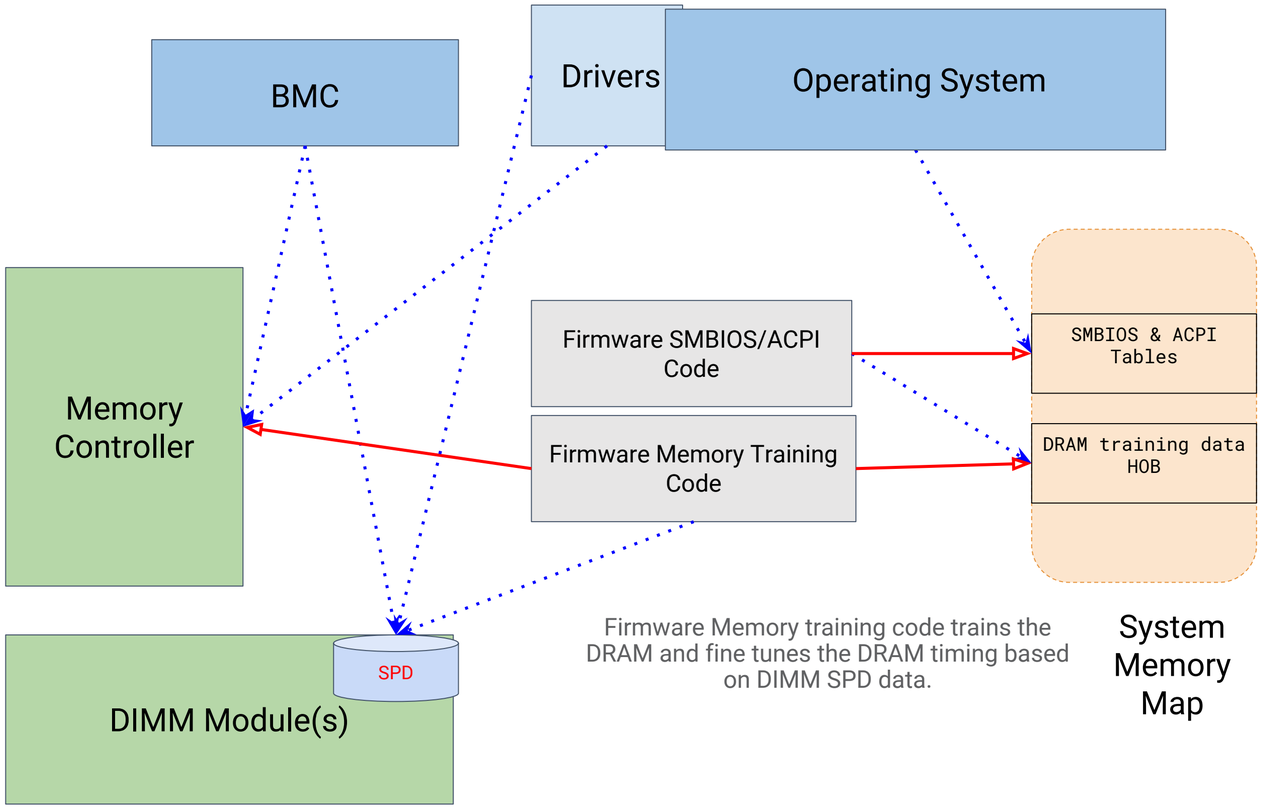


- MRC creates Hand-Off Blocks for higher firmware stack with all the relevant information
- The HOB info is exported to OS in various standard tables like SMBIOS/ACPI etc.

DIMM Software Stack: OS & BMC

OPEN SYSTEMS FIRMWARE

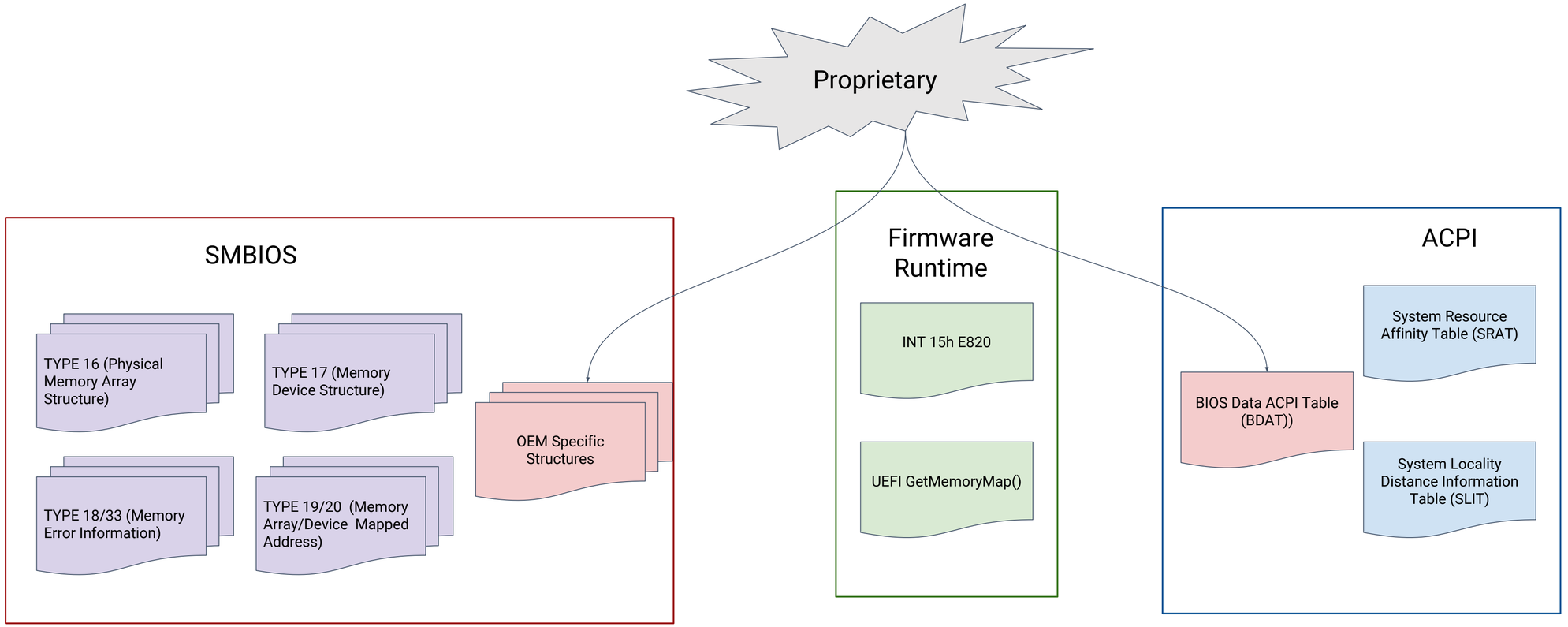
BMC thru inband or outband, queries DRAM data for FRU and sensor management.



- For system memory map the OS relies on standard, Firmware exported data like ACPI tables
- For repair workflow (for DIMM) and other timing info OS/BMC has to use proprietary methods even though underlying technology (DDR3/DDR4 etc) is standard

Firmware Memory Data Interfaces

OPEN SYSTEMS FIRMWARE



Background: DIMM Training Data

OPEN SYSTEMS FIRMWARE

- As we can see there is no standard way to expose the critical DRAM Training data to OS and higher software layer
- Why we need it to expose it higher layer?
  - It helps in Data center operations where the holistic view provides more valuable data and better decisions can be made.
    - (like) “If you want to run something reliable with predictable performance, avoid that system in that corner”
    - (like) “Change the DIMM #8 in that corner system, it might fail - I saw more frequent errors over there”

SMBIOS Structures

OPEN SYSTEMS FIRMWARE

- Memory device (Type 17) structure provide info on individual DIMM such as
  - DIMM dimensions size, width, speed, voltage etc
  - DIMM identifier such as location string, asset tag, serial number etc
  - Handle to error information structure (Type 18 or 33)
  - Handle to the physical memory array group
  - Usually not filled in if the DIMM is disabled due to training error
- Physical memory array (Type 16) structure groups memory devices to logical unit say a channel or memory riser etc.
- Memory array mapped address (Type 19) & memory device mapped address (Type 20) structures provide ability to translate memory address to DIMM
- As SMBIOS allows variable structure size, vendors use the extra data for storing proprietary info about the DIMM

ACPI Tables

OPEN SYSTEMS FIRMWARE

- Static Resource Affinity Table (SRAT)
  - Describes the resource affinity domains in the system
  - Critical structure to understand the Non-Unified Memory Architecture (NUMA) of the system
- System Locality Distance Information Table (SLIT)
  - Describes the locality of NUMA (Non-Unified Memory Architecture) nodes
  - The distance is given in terms of Memory Latency

Filling the gap

OPEN SYSTEMS FIRMWARE

- DRAM technologies are pretty much standard like DDR3, DDR4 and they have clear specifications like what goes in SPD (Serial Presence detect)
- Unfortunately there is no standard related to DRAM timing and training data

ACPI BDAT & Schemas

OPEN SYSTEMS FIRMWARE

- BDAT is an ACPI table with standard ACPI header etc.
- BDAT consists of schemas
- Schemas are basically UUID (GUID) based data structure
- The UUID identifies the type of data associated with the schema
- Multiple schemas will be arranged together to form a coherent data structure
- BDAT schemas provide flexibility in defining new data formats

Proposed Open Firmware Data

OPEN SYSTEMS FIRMWARE

- Start from Intel’s BDAT structure or something similar. Reasons?
  - Already used in multiple platforms
  - Vendors have experience working with it
- Gotchas:
  - In its current format the structures are wasting lots of space and it is hard to parse.
- Proposal is to start from existing BDAT and modify existing or define new schema

Proposed Action: Open Source Kernel Driver

OPEN SYSTEMS FIRMWARE

- Develop a kernel driver (bdat) to expose BDAT schemas
- Create necessary /sys nodes to access BDAT data in userspace

Call to Action

OPEN SYSTEMS FIRMWARE

- Work as a community to make it available in all platforms
- Provide feedback and add missing data
- Create relevant support in Open Source community like standard Linux drivers and tools
- Create additional interfaces like RedFish/IPMI to uniformly identify and utilize this data


Open. Together.
OCP Global Summit | March 14–15, 2019
OCP
SUMMIT