OSF on Intel Xeon processor-based Servers from Blueprint to Production Quality
OSF on Intel Xeon processor-based Servers from Blueprint to Production Quality

Anjaneya "Reddy" Chagam, Cloud Architect, Intel
Jonathan Zhang, Software Engineer, Meta
Intel technologies may require enabled hardware, software or service activation. No product or component can be absolutely secure. Your costs and results may vary. © Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

For workloads and configurations visit www.Intel.com/PerformanceIndex. Results may vary.

Intel technologies may require enabled hardware, software or service activation. © Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

Includes the effect of Intel Thermal Velocity Boost, a feature that opportunistically and automatically increases clock frequency above single-core and multi-core Intel Turbo Boost Technology frequencies based on how much the processor is operating below its maximum temperature and whether turbo power budget is available. The frequency gain and duration is dependent on the workload, capabilities of the processor and the processor cooling solution.

Code names are used by Intel to identify products, technologies, or services that are in development and not publicly available. These are not "commercial" names and not intended to function as trademarks.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.
Agenda

• The path till today
• How did we get here
• Benefits demonstrated
• Next steps
Agenda

• The path till today
• How did we get here
• Benefits demonstrated
• Next steps
The OSF and Xeon-SP

(Tux the penguin: lewing@isc.tamu.edu)

(LinuxBoot logo: https://www.linuxboot.org/page/artwork/)
## The Path

<table>
<thead>
<tr>
<th>OCP Server</th>
<th>Xeon Server Processor</th>
<th>FSP Status</th>
<th>Work Duration</th>
<th>Test Pass Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>TiogaPass</td>
<td>SkyLake - SP</td>
<td>Prototype</td>
<td>Jan. 2019 - Feb. 2020</td>
<td>38%</td>
</tr>
<tr>
<td>DeltaLake</td>
<td>CooperLake - SP</td>
<td>Statement Of Work</td>
<td>Mar. 2020 - July 2021</td>
<td>95%</td>
</tr>
<tr>
<td>Next Generation single socket server</td>
<td>SapphireRapids - SP (SP-SP)</td>
<td>Included in Intel Reference Platform BIOS</td>
<td>Jan. 2021 on-going</td>
<td>Trending better than DeltaLake</td>
</tr>
</tbody>
</table>

- **Single** coreboot code base: [https://review.coreboot.org/](https://review.coreboot.org/)
- **Single** LinuxBoot kernel code base: [https://www.kernel.org/](https://www.kernel.org/)
- **Single** LinuxBoot initramfs code base: [https://github.com/u-root/u-root/](https://github.com/u-root/u-root/)
## The Lookback

<table>
<thead>
<tr>
<th></th>
<th>2018</th>
<th>2021</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FSP</strong></td>
<td>No FSP for Xeon-SP</td>
<td>FSP for SPR-SP is POR</td>
</tr>
<tr>
<td><strong>coreboot</strong></td>
<td>No support for Xeon-SP and server features</td>
<td>Support for Xeon-SP and server features is mature</td>
</tr>
<tr>
<td><strong>LinuxBoot</strong></td>
<td>No support for bootloader</td>
<td>bootloader technology is mature</td>
</tr>
<tr>
<td><strong>Ecosystem</strong></td>
<td>Not exist</td>
<td>Industry coalition formed to collaborate on OSF for Xeon-SP; Partial stack deployed by multiple hyperscalers, some deployments are at scale</td>
</tr>
</tbody>
</table>
2019

- Open System Firmware from scratch
- OCP regional summit 2019
- Jonathan Zhang / David Hendricks

Agenda

- What is FB doing? Why?
- The build system design
- How do we enable our partners and the community?
2020

- Xeon-SP FSP and coreboot status update
- OCP global virtual summit 2020
- Anjaneya Reddy Chagam, Intel
- Jonathan Zhang, Meta

Agenda

- Facebook Open System Firmware
- Intel® Firmware Support Package (FSP) Overview
- coreboot Xeon SP (SKX & CPX) FSP status
- Summary and Next Steps
- Demo
2020

- OCP SErver Feature Development with coreboot/Linuxboot
- OCP global virtual summit 2020
- Jonathan Zhang, Meta
- Johnny Lin, Wiwynn

Agenda

- Overview
- Server Features and Designs
- Call to Action
OCP Delta Lake Open System Firmware

The open system firmware contribution to OCP Delta Lake is hosted in this directory. OCP Delta Lake is a single socket server design based on the 3rd Gen Intel Xeon Scalable processors, which is referred to as Delta Lake IS Server. OCP Delta Lake server is a component of multi-node server system Yosemite-V3. The open system firmware solution developed for Delta Lake is based on coreboot/LinuxBoot stack. The support status and information of coreboot/LinuxBoot for OCP Delta Lake can be found here. Please note that the support status described in the link was developed and validated against an engineering Intel FSP which is not public, but it shouldn’t be a huge gap to work on the current upstream coreboot to reach the same status with the public Intel FSP. If you need professional service for this effort or other features development, please see Maintainers and Professional Support. The files in this directory can help you download all the public source code and binary files, and build the final system firmware image that can pass OCP OSF checklist. In the patch file in src directory, config-deltalake.json configures all the open source upstream repositories and commit hashes, including the toolchain that will be downloaded and built. For more detail about the fetching and building process please see How To Build.

Build Pre-requisites

- Linux OS
- GNU Make
- python

How To Build

cd Wiwynn/deltalake && ./download_and_build.sh
DeltaLake As OCP Accepted:
Open Hardware + Open Firmware = Happier OCP Community
Agenda

• The path till today
• **How did we get here**
• Benefits demonstrated
• Next steps
Intel Firmware Support Package (FSP)

- FSP spec publicly available
- Distributed as binary after platform launch
  - CooperLake/IceLake Xeon Processor FSP binaries available now on Intel github
- Well defined interfaces for boot loader integration (e.g., coreboot)
  - OCP Tioga Pass (Skylake-SP), DeltaLake (CooperLake-SP) coreboot firmware available now
- Binary customization
- Unmet needs that require focus
  - RAS support using Standalone MM
Upstream First

• Use one code base for all platform
• Insist quality code update
• Uphold the design philosophy and architecture
• Fill in gaps:
  ○ coreboot -- Xeon-SP split IIO, RAS, CBnT, BMC interaction, configuration management etc.
  ○ u-root -- bootloader and tools (system boot)
  ○ Test harness: https://github.com/facebookincubator/contest
  ○ Build system: https://github.com/facebookincubator/osf-builder/
Ecosystem

• Solve industry problems
  ○ OSF for MonoLake serving circular economy
  ○ OSF for DeltaLake enabling OCP adoption

• Form industry coalition
  ○ No reinventing the wheel
  ○ Collaborating on design, debug and code review on technologies of common interest
Test, Test, Test

• Power/Performance
• Register level check: Intel self test tool
• Industry standard tests: FTWS
• Solution level tests: tests from OS, from OpenBMC
• Stress tests
Agenda

• The path till today
• How did we get here
• **Benefits demonstrated**
• Next steps
New technology Development

• End to end solution involves more than host firmware
  ○ Hardware (Silicon, platform)
  ○ Host firmware
  ○ kernel/OS
  ○ Device Firmware
  ○ Infrastructure

• Distributed knowledge
  ○ Hype-scaler: solution level requirement
  ○ Silicon Vendor: processor internal
  ○ Device Vendor
  ○ Server Vendor

OSF enables shorter TTM for new technology
Case study: CXL Memory Device

• **Challenge**
  For CXL memory device with externally attached DIMM, how to populate smbios tables?

• **Reality**
  ○ The CXL spec does not provide a way for CXL device to communicate this to host
  ○ This is not part of silicon initialization

• **Conclusion**
  ○ OSF to provide reference implementation
Server Platform Development

- Coreboot makes it easier to debug issues in other components
  - Hardware design
  - Other firmware components
- Linuxboot enables easier debugging of kernel related issues.

OSF architecture helps with quality server platform development
New Server Development

- **Engineering philosophy**
  - Single code base
    - for different platform (such as multiple generations of Xeon-SP processors)
    - among partners
  - Partner/Community collaboration
    - Bug fixes
    - New features
    - Code review

- **Benefits**
  - Lowered development cost
  - Better quality
Sustaining

- Linux engineer as firmware engineer
- One set of device driver for both target OS and firmware → improved quality and security
- One set of infrastructure tools for both target OS and firmware → simplified infrastructure tools
- Open code base instead of closed code base

OSF reduces problem surfaces and enables faster problem solving
Agenda

• The path till today
• How did we get here
• Benefits demonstrated
• Next steps
Next Steps

• Technology
  ○ Follower → Leader

• Ecosystem
  ○ Production Ready → Industry Adoption
Call to Action

- Join us in this transformation
  - OCP OSF project: https://www.opencompute.org/projects/open-system-firmware
  - Open source communities:
    - coreboot: https://www.coreboot.org/
    - Linuxboot: https://www.linuxboot.org/
    - contest: https://github.com/facebookincubator/contest
    - osf-builder: https://github.com/linuxboot/osf-builder/

- Benefit from OSF powered solution
  - OCP for OCP DeltaLake Server:
    https://github.com/opencomputeproject/OpenSystemFirmware/tree/master/Wiwynn
  - OCP MonoLake Server:
    https://review.coreboot.org/c/coreboot/+/57561
Open Discussion