Tianocore Update

Leif Lindholm (ARM/Linaro)
## Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UEFI</td>
<td>A specification of interfaces which need to be provided by firmware and which portable applications and drivers can be written against.</td>
</tr>
<tr>
<td>PI</td>
<td>Platform Initialization. A specification also maintained by the UEFI forum, describing internal interfaces in a firmware implementation.</td>
</tr>
<tr>
<td>EDK2</td>
<td>A UEFI/PI implementation maintained as part of the Tianocore project.</td>
</tr>
<tr>
<td>Tianocore</td>
<td>An opensource project encompassing EDK2 and other related software components.</td>
</tr>
<tr>
<td>Ovmf</td>
<td>EDK2 platform port to virtual machines (QEMU, Xen)</td>
</tr>
</tbody>
</table>
Tianocore Governance

- One of the things I have brought up in the past is how lack of any sort of formal leadership in Tianocore has been holding back and slowing down necessary changes
  - The migration to GIT finally happened at the start of this year, but it took a long, long time to get there
  - The question of how to integrate more open-source device drivers and platform support remains unsolved (but more later)

- As of March this year, Tianocore now has three *stewards*, empowered to drive consensus
  - Andrew Fish (Apple)
  - Mike Kinney (Intel)
  - Me
EDK2 FAT Filesystem Driver

- Was always non-free software, covered by a different license than the rest of EDK2
  - Banning use outside of EDK2

- Apart from the bad PR, this made most Linux distributions unable to ship Ovmf images as part of their main archive
  - And as a result, most cloud infrastructure tools initially supported only SeaBIOS.
  - So surprisingly, some of the initial enablement for AArch64 required work on UEFI support (or at least pulling the very latest upstream).

- But back in March/April, Microsoft agreed to relicense under default (2-clause BSD)
  - Now able to be included directly in main repositories
OpenPlatform Pkg

- Since last Connect, we have added support for new platforms
  - AMD Seattle
    - Husky/Cello
    - Overdrive
  - Hisilicon
    - D02/D03
  - Marvell
    - Armada 70x0

- We have also published the resurrected Marvell Yukon Ethernet driver
  - And with some help from SoftIron, debugged it across at least two fundamentally different systems

- But hopefully I won’t be talking about it at next Connect...
Proposals

- A mechanism to formally drive change in the project. Currently live ones are
  - Revamp of the EDK2 directory structure
  - Defining the mechanism by which new platform code and drivers are introduced into EDK2

- Directory structure
  - Plan is to drop the outdated “IP silo” model with “packages” (*Pkg)
  - Define some more obvious locations to place device drivers under

- Platform code
  - Main EDK2 repository to contain only enough platform code to validate the core code
  - Separate platforms repository to be set up to hold others
EBC Support

- EFI Byte Code - enables architecture-independent device drivers, including option ROMs.
  - AArch64 support implemented by Code Aurora, copyright assigned to the Linux Foundation.
  - Some fixes and improvements by Ard.

- Currently only produced by a Non-Free compiler from Intel
  - Working only under Windows + Visual Studio
  - And many distributors will try to convince you it does not exist.
  - Based on member interest, we’ve worked with the toolchain group to draft a plan for an LLVM-based alternative.

- Microsoft currently not signing EBC drivers
  - Interested parties need to speak up in the UEFI forum
SCT

- The UEFI Self-Certification Testsuite is released periodically by the UEFI forum, but unlike EDK2, it has never transitioned to an open development model
  - Code available on github to registered UEFI forum members
  - Main contributors are Intel (90%) and ARM (9%?)
  - Not much in the way of public discussion of upcoming changes

- Some proposals made during UEFI Plugfest last week
  - Create a development mailing list (access restricted for now)
  - Consider breaking out the test framework as a public-access separate project from the tests themselves (while investigating the possibility of opening the tests up longer-term)
This Slide Intentionally Left Blank
Honorary Mention

Alexander Graf keeps improving the bootefi support for U-Boot.

Even though the implementation leaves out large bits of UEFI functionality, an alternative codebase is a very useful tool for us to use for validation. And to improve our validation.

I want to:

● Run the UEFI shell on U-Boot
● Run SCT on U-Boot
And One More Thing...

On 21 September 2016 at 08:09, Matt Fleming <matt@codeblueprint.co.uk> wrote:

> Folks,
>
> I've asked, and Ard has agreed to step up and help me co-maintain the
> EFI subsystem.
>
> Given that there are now two maintainers, we're moving to a shared git
> repository on kernel.org, hosted at,
>
> git://git.kernel.org/pub/scm/linux/kernel/git/efi/efi.git
>
> Expect a MAINTAINERS patch soon.
>
> I do plan on keeping the existing tree in sync for the time being, so
> it won't actually matter which repository people base their patches
> on. Hopefully the disruption to patch submitters will be minimal.
>
> Thanks again Ard!
Thank You

#LAS16

For further information: www.linaro.org
LAS16 keynotes and videos on: connect.linaro.org