



A Quick History of UEFI Networking

May 2013

Vincent Zimmer



Background:

History of in-band firmware/BIOS networking

What is PXE?

Platform firmware has had integrated network boot capabilities since the Pre-Boot Execution Environment (PXE) was defined in the mid 1990's as part of the Wired For Management (WFM) efforts. PXE can be thought of as 'in-band' networking since it runs on the main host CPU's, not an out-of-band chipset or platform microprocessor, or a 'non-host processor' (NHP). PXE entails two elements – (1) client side API's, including Base-Code and UNDI, for purposes of orchestrating the download of a network boot program (NBP) and (2) a wire protocol for interacting with the boot server. The PXE boot process is 'client initiated' in that the in-band firmware initiates a DHCP discovery process to start the networking interactions with a boot server.

Where was PXE defined?

PXE was originally part of the PC/AT BIOS and an specification jointly owned by a small consortium of companies. With the advent of the Extensible Firmware Interface (EFI) in the late 1990's, the 'base code' and 'UNDI' interfaces from BIOS PXE were mapped into EFI interfaces. This continued through the EFI 1.10 specification in 2001. The EFI1.10 specification was an Intel-owned document. In order to support broader industry adoption, EFI 1.10 was contributed to the Unified Extensible Firmware Interface (UEFI) forum in 2005, along with some post EFI1.10 networking API's. The latter included a modular IPV4 network stack that broke out IP, UDP, TCP, DHCP, ARP and other elements into separate API's, as opposed to the EFI1.10 reference implementation of the monolithic PXE stack. With this modular network stack in UEFI2.0 in 2006, the foundation was laid to create additional networking services on the UEFI platform. These services have included a refactored PXE Client that leverages the modular network stack and an iSCSI initiator, both in the open source.

More information on UEFI can be found at [\[BEYOND-BIOS\]](#) and the specification itself at [\[UEFI\]](#).

Post-PXE?

The question was posed to the industry group in 2007 about how to evolve PXE. At the time, there were many extant scenarios built upon the IPV4 PXE wire protocol, including support in all of the Linux distributions and the Windows Deployment Services (WDS) feature in Microsoft Windows. The most important feature request entailed addition of IPV6 support. As such, the UEFI Forum worked w/ the IETF and generated RFC 5970 that includes the option tags for IPV6 network boot. This RFC, along with a network interaction flow, form 'netboot6', or a variant of PXE that interoperates across IPV6.



The IETF and the UEFI Network Subteam (UNST), chaired by Vincent Zimmer [ITJ-BB], evolve the pre-OS wire protocols and UEFI API's, respectively.

The modular network stack and the IPV6 and IPV4 variants of PXE [PXE-SPEC] can be found in the EFI Developer Kit 2 (edk2) project on source forge (edk2.sourceforge.net) in the Network Package (NetworkPkg) [EDK2]. Some details of the packages can be found below in Figure 1.

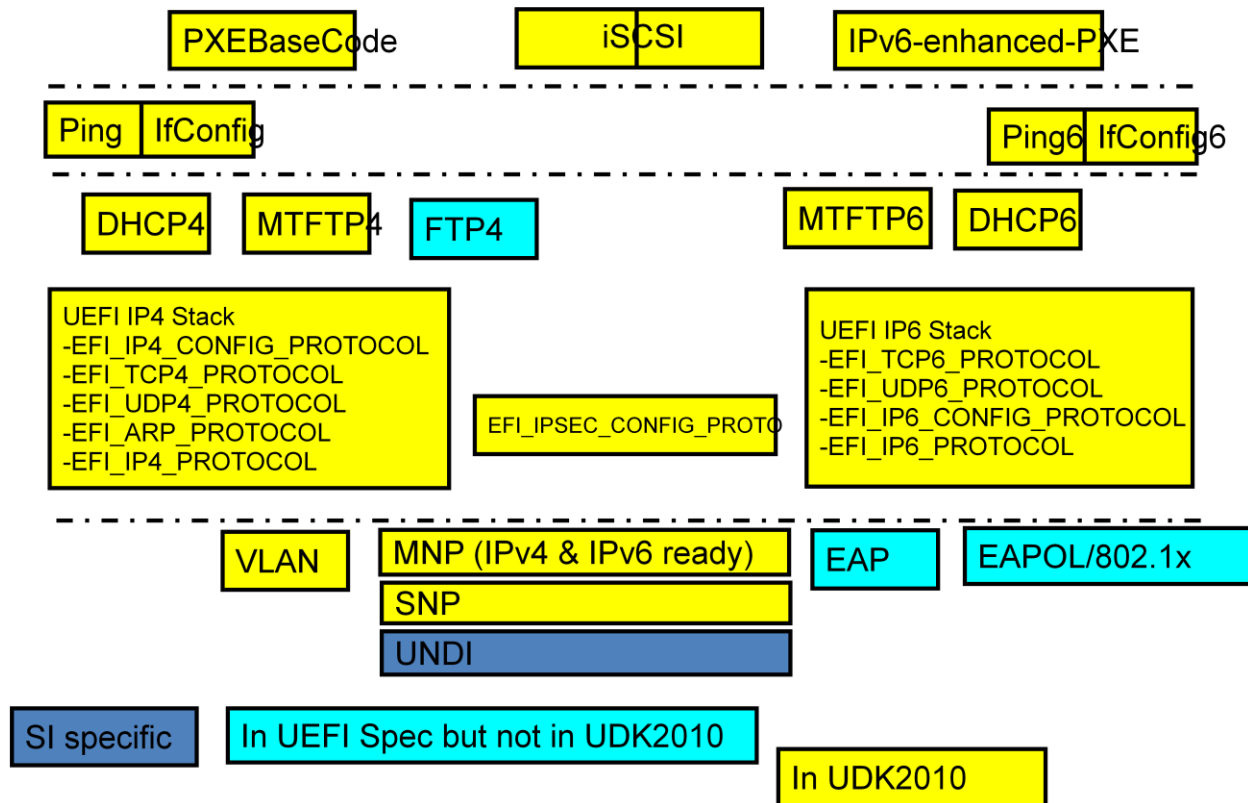


Figure 1 Network stack layout

Netboot6, along w/ UEFI Secure Boot [ITJ-SB], provided UEFI-only features not available on a PC/AT BIOS. These features, along w/ fast-boot, were integrated into Microsoft Windows8 and helped motivate the decision to mandate UEFI 2.3.1c specification conformance for this operating system in 2012. So the EFI effort that commenced in 1998 culminated in the 2012 launch of Windows8 that required UEFI.

This was the tipping point for the standards adoption and many of the preceding tactics, such as feature creation and open source, helped to motivate this decision.



Conclusion:

The UEFI standards and open source provide a foundation upon which many capabilities can be deployed.

References:

[BEYOND-BIOS] Zimmer, et al *Beyond BIOS: Developing with the Unified Extensible Firmware Interface*, Second Edition, November 2010

<http://noggin.intel.com/intelpress/categories/books/beyond-bios-2nd-edition>

[EDK2] EFI Developer Kit II <http://edk2.sourceforge.net>

[ITJ-BB] Mark Doran, Vincent Zimmer, Michael Rothman, "Beyond BIOS: Exploring the Many Dimensions of the Unified Extensible Firmware Interface," in *Intel Technology Journal - UEFI Today: Bootstrapping the Continuum*, Volume 15, Issue 1, pp. 8-21, October 2011, ISBN 978-1-

934053-43-0, ISSN 1535-864X <http://noggin.intel.com/technology-journal/2011/151/uefi-today-bootstrapping-continuum>

[ITJ-SB] Magnus Nystrom (Microsoft), Martin Nicholes (Insyde), Vincent Zimmer, "UEFI Networking and Pre-OS Security," in *Intel Technology Journal - UEFI Today: Bootstrapping the Continuum*, Volume 15, Issue 1, pp. 80-101, October 2011, ISBN 978-1-934053-43-0, ISSN 1535-864X <http://noggin.intel.com/technology-journal/2011/151/uefi-today-bootstrapping-continuum>

[PXE-SPEC] Pre-Boot Execution Environment 2.1

<http://download.intel.com/design/archives/wfm/downloads/pxespec.pdf>

[RFC] T. Huth (IBM Germany), J. Freimann (IBM Germany), V. Zimmer (Intel), D. Thaler (Microsoft), "DHCPv6 Options for Network Boot," Internet RFCs, ISSN 2070-1721, RFC 5970, September 2010, <http://www.rfc-editor.org/rfc/rfc5970.txt>

[UEFI] Unified Extensible Firmware Interface <http://www.uefi.org> UEFI Specification, Revision 2.3.1c, UEFI Platform Initialization (PI) Specification, revision 1.2.1