**Say Goodbye To Your BIOS: Hello, UEFI!**

By Patrick Schmid and Achim Roos DECEMBER 29, 2009 11:00 PM

1. **On the verge of UEFI**

The days of the good ol’ BIOS are numbered. The Unified Extensible Firmware Interface (UEFI) will introduce a more powerful solution able to better cope with the demands of today’s diverse hardware. In a nutshell, UEFI is an interface that takes care of handing over the pre-boot environment to the operating system. We took a quick look at UEFI and found some imminent issues.

You may have already heard about UEFI (or EFI, which was the initial approach). Intel initiated EFI in 2003 with the Itanium’s IA64 architecture under the title Boot Initiative. The concept was handed over to the Unified EFI Forum, which managed and promoted the new standard for the entire industry. AMD, AMI, Apple, Dell, HP, IBM, Insyde, Intel, Lenovo, Microsoft, and Phoenix are the leading members today. A BIOS relies on the x86 architecture’s 16-bit real mode, but UEFI introduces full hardware independence and interfaces split into boot and runtime services. These aim at high standardization while introducing enough flexibility for manufacturers to differentiate their products.

**The BIOS: Antique and Still Prevalent**

Some 25 years ago, the BIOS was designed to launch operating systems. The first computers used punch cards as a launch target before these were replaced by ROMs with basic interpreters. Today we can chose a plethora of targets, including floppy disks, hard drives, optical drives, and network locations. However, actual component operation is still subject to the particular operating system through device drivers, whereas EFI allows for OS-independent driver support through its own driver model.

There have been several attempts to modify the initial BIOS concept. IBM introduced a modified system design, the PS/2, in 1988, partly to fight off BIOS clones. Its 32-bit Multi Channel Architecture (MCA) with ABIOS can be seen as a way around copycat BIOS implementations. Advanced RISC Computing (ARC) aimed at uniting the boot environments of the MIPS and Alpha platforms in the 1990s, but it lacked an evolutionary path, extensibility, and possible system diversity. PowerPC and SPARC have their Open Firmware (OF) and Common Hardware Reference Platform (CHRP), which sort of worked against ACPI by not embracing it.

In the end, the BIOS is still here and still does what it has been doing for the last 25 years: making sure your operating system can boot. It was never designed for today’s massive diversity of hardware. It’s still stuck with 16-bit interfaces and software interrupts, interrupt routing and maximum precision timers, limited ROM execution space (1 MB) and image size, a limited number of initializable devices (which is critical in the server space), proprietary extensions, and missing modularity—just to name a few issues.

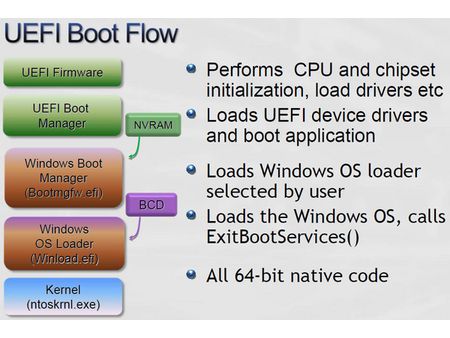
**UEFI Support**

Operating systems started to support the platform interface design by 2007, but most Windows versions, such as Vista with SP1 and Windows Server 2008, only offered support on the 64-bit editions. Unfortunately, we found that industry support for UEFI is still very weak, and there are some shortcomings on the storage end.

1. **What UEFI can do for you**

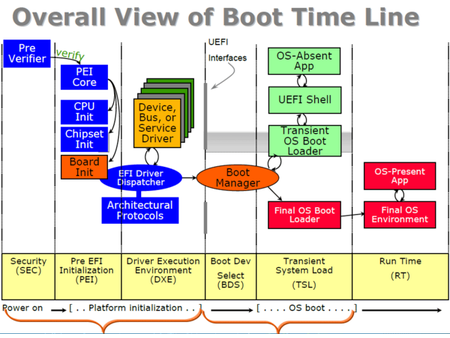
EFI is a software interface that operates between the operating system and the platform firmware, allowing it to be capable of replacing the BIOS. As a matter of fact, EFI is complex enough to be called “OS-like.” It not only offers driver support, interfaces, and services, but it also has a shell that allows users to execute applications on a command line interface.

EFI contains system information organized into tables, and it offers boot and runtime services to the operating system. Boot services include initialization, file services, and textual and graphical user consoles. Runtime services include date, time, and NVRAM services. To enable or facilitate inter-device communication, all EFI drivers and components communicate through specified protocols. Drivers are also important, since the EFI device driver environment, the EFI Byte Code (EBC), is a processor-independent environment that allows for both device initialization and operation (with the exception of OS-specific hardware with specific feature and driver support requirements, such as high-end graphics cards).

[[](http://www.tomshardware.com/gallery/uefi-boot-flow,0101-231056-0-2-3-1-jpg-.html)](http://www.tomshardware.com/gallery/uefi-boot-flow,0101-231056-0-2-3-1-jpg-.html)

Gone are the days when you needed boot manager software to start different operating systems. EFI implements its own boot manager to take care of this job. System builders have new opportunities as well, because EFI extensions may be added from various, non-volatile locations. This means that a PC maker could use the customer’s desired motherboard and platform while adding additional system features via EFI through a partition of the system hard drive or, eventually, even through Internet downloads.

Apple’s Boot Camp, which allows users to install a Windows version next to an existing Mac OS installation in a non-destructive manner, actually utilizes Intel’s EFI implementation on Intel-powered Macs instead of Open Firmware. This has been available since April of 2005 and is not  nearly as easily to pull off on x86 PC platforms.

[](http://www.tomshardware.com/gallery/uefi-boot,0101-231055-0-2-3-1-png-.html)

Since EFI’s Platform Initialization (PI) feature offers a Compatibility Support Mode (CSM) through which it can publish additional interfaces, it’s also possible to provide a legacy BIOS runtime. Given the benefits, this makes it very appealing for manufacturers to adopt UEFI quickly. However, today’s broad support exists mostly on paper.

1. **Industry support? Fail**

[Wikipedia’s EFI page](http://en.wikipedia.org/wiki/Extensible_Firmware_Interface) shows a list of EFI-toting systems in 2008 from IBM, HP, and HP Compaq, adds more systems for 2009 from IBM and Dell, and refers to systems mentioned in an UEFI Forum document. The 2008 list also mentions MSI motherboards with ClickBIOS. Since we’re closer to the component [companies](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-3.html) than to system builders, we decided to look around and get a feeling on where the big motherboard vendors are with their EFI ambitions.

**Asus:**

“We currently offer an EFI solution for the P5Q Deluxe, but it is still in beta status. It will depend on public demand whether or not we add EFI updates for additional platforms.”

Asus’s EFI [software](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-3.html) for the P5Q Deluxe [is available here](ftp://ftp.asus.com.tw/pub/ASUS/mb/socket775/P5Q_Deluxe/EFI/). A thread on the [Asus forums](http://vip.asus.com/forum/view.aspx?SLanguage=en-us&id=20091019202824968&board_id=1&model=P5Q%20Deluxe&page=1&count=32)still contains a lot of bug reports on this, though.

**Foxconn:**

“Currently, our boards do not use the Unified Extensible Firmware Interace feature and we don’t have products under development utilizing this type of BIOS.”

Gigabyte:

“We’re working on possibly bringing UEFI to our products, but we don’t have a timeframe yet.”

**Intel:**

“An increasing number of platforms will be outfit with UEFI, rather than a conventional BIOS.”

However, we found that Intel introduced limitations to partitions larger than 2TB. Although UEFI could have an operating system boot from such a large partition when using the GPT (GUID Partition Table) instead of the MBR, Intel says that the “RAID option ROM will only allow a RAID volume to be bootable if it is < 2TB.”

|  |  |
| --- | --- |
| http://media.bestofmicro.com/UEFI-firmware-BIOS,A-2-231050-13.jpg | http://media.bestofmicro.com/UEFI-firmware-BIOS,A-3-231051-13.jpg |

**MSI:**

“MSI shipped an UEFI-enabled platform with ClickBIOS based on the P45 chipset, but the product has reached end of life.”

**Moving On**

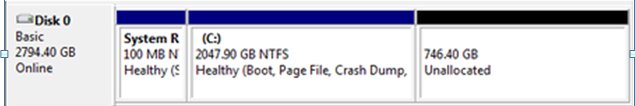
We understand that this area is still new, but it appears as if the key motherboard makers aren’t very receptive to the obviously desirable system interface. The processes behind these decisions are undoubtedly complex, and we don’t want to overrate this yet. But we also believe that it’s about time for the industry to seriously look at making the switch. There should at least be a clear commitment to making certain platforms upgradeable to UEFI when necessary. Otherwise, I wouldn’t want to buy an expensive motherboard knowing that support might not last long.

EFI-based features, such as integrated overclocking functionality, might still have to be developed, so we decided not to look at different motherboards with UEFI support and make judgements based on premature or hesitant solutions. But there is one scenario that will actually require UEFI soon: **if you want to boot from a partition that is larger than 2TB, you positively need UEFI and GPT.**

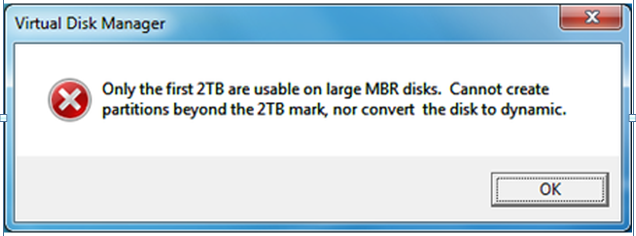
Storage appliances that potentially exceed 2TB capacity by utilizing RAID [technology](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-3.html) already cross this line when used as eSATA-based boot volumes. Therefore, we decided to pick one of the more [advanced](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-3.html) UEFI implementations and look at whether or not we could work on it using high-capacity system drives.

1. **Storage Partitioning: The Issues With MBR**

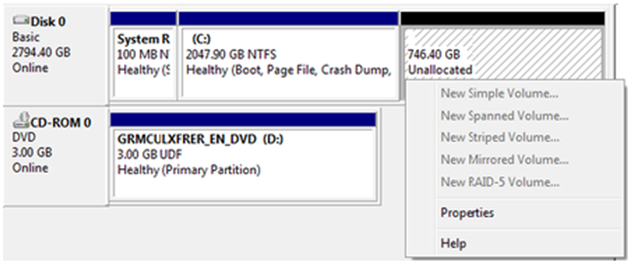
Windows setup automatically selects GPT for volumes larger than 2TB and allows users to decide between MBR and GPT for capacities under 2TB. This also applies to the disk management utility if you manage drive volumes on an existing Windows installation. Many BIOSes available today support GPT, but a system without UEFI support wouldn’t be capable of booting an operating system from a GPT partition once it is larger than 2TB. Let’s look at what happens if you go with the conventional MBR.



If UEFI support is unavailable, Windows will still automatically use GPT for drives 2TB or larger and offer the choice between GPT and MBR. All unused capacity above 2TB will not be used (and remain useless if partitioned with MBR).



Windows Vista is clear on this.



You would think that it should be possible to create an additional partition once a 2TB volume has been installed on a drive with 2TB or more of capacity. This is not possible, as MBR limits the entire drive to one partition of up to 2TB. The remaining storage space is now useless. Hence, it is important to only use GPT for large capacity drives, as you won’t face the 2TB limit.

So let’s move on and create a large partition using the GPT layout and the LaCie 4big Quadra eSATA drive.

1. **Storage Partitioning With GPT, And MBR With ESP, MSR**

You need A 64-bit version of Windows (XP, Vista, or 7) to get full GPT support for drives that offer more than 2TB capacity. GPT offers the following features:

* 18 exabyte maximum capacity
* Up to 128 partitions per disk
* Primary and backup partition tables available for redundancy
* Well-defined and self-identifying partition format
* Each partition has a unique ID to avoid identifier collisions (“GUID” partition table)

Here is an overview of your possible partitioning options and actions on GPT and MBR:

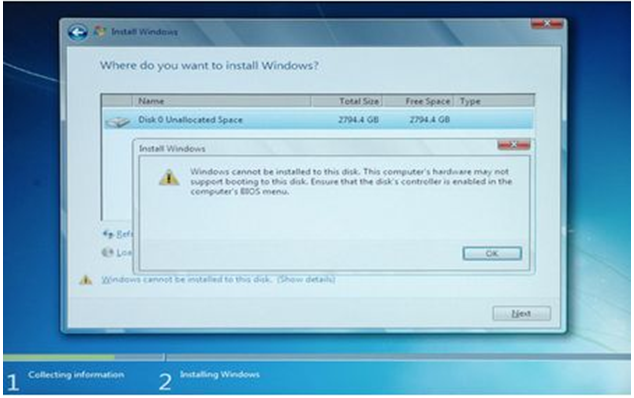
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 32-bit Windows | | 64-bit Windows | |
|  | **GPT** | **MBR** | **GPT** | **MBR** |
| **Boot** | No | Yes | Yes | No |
| **Read** | No | Yes | Yes | Yes |
| **Write** | No | Yes | Yes | Yes |

Keep in mind that UEFI support becomes relevant for drives >2TB. It must be present to enable booting from such large partitions provided the other prerequisites (64-bit Windows and GPT) are met.

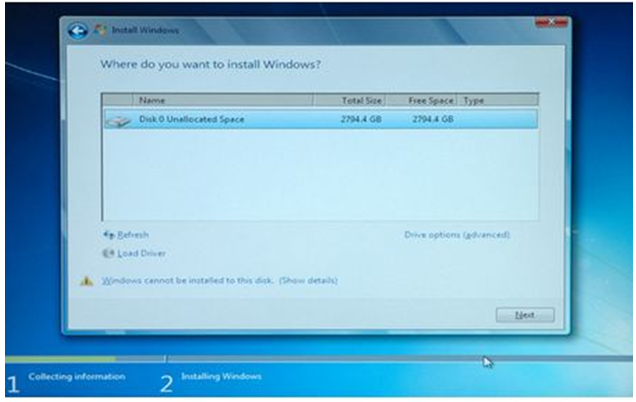
**GPT Details on Windows x64**

GPT will automatically install the EFI System Partition (ESP), which contains the boot loader, EFI drivers, and all other necessary files for booting the system (such as boot.ini, HAL, and NT Loader). It utilizes the GUID Partition Table rather than the master boot record. The ESP is 1% of the drive capacity, or a minimum size of 100MB and a maximum size of 1,000MB.

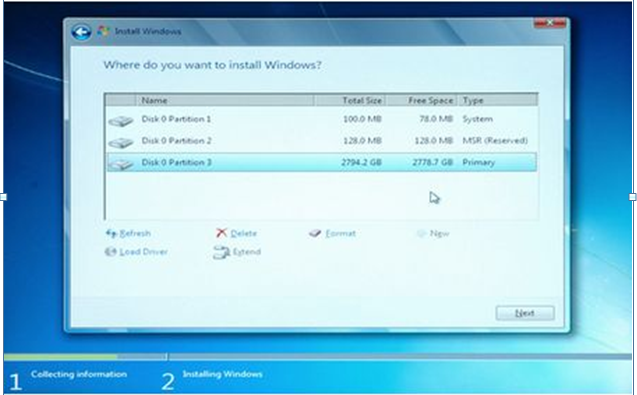
GPT systems will also be equipped with an MSR partition, which stands for Microsoft Reserved. Since GPT partitions don’t allow hidden sectors, Windows utilizes this reserved space for operating system use. If you decide to convert a basic disk into a dynamic disk, Windows will use the MSR partition, shrink its size, and create the dynamic disk[database](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-7.html) using available space. On drives smaller than 16GB, the MSR partition will be only 32MB. For larger disks, it will consume 128MB.



Even if your system is GPT-compliant and created using the necessary partition. you will still need UEFI support to be able to boot from a large volume.



You can go ahead and still install Windows onto the unbootable partition, but you were warned.



If GPT and UEFI support are both working properly, Windows 7 will automatically create the necessary partitions, such as the 100MB EFI System Partition (ESP) and the 128MB Microsoft Reserved partition (MSR), and then let you create a high-capacity volume that is indeed bootable.

1. **Conclusion**

The industry has come a long way with UEFI. There have been various approaches to modularize the boot process and create a flexible standard. None of them were a real success, but it seems that Intel may have won at least some support with its ambitious Itanium project. EFI, which today is governed by the United EFI Industry Forum, controls the existing UEFI standards. The Extensible Firmware Interface will gradually replace the conventional BIOS and offer new interfaces to operating systems, facilitating boot-up and introducing a lot of flexibility through EFI applications and introducing a lot of flexibility through EFI applications and OS-independent device drivers.

Although the recent 2.3 specification is mature, the industry has not yet adopted the new standard. From an end-user’s perspective, this is difficult to understand. Apple, [IBM](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-8.html), HP, and a few others have already proved that UEFI can indeed be implemented into production systems. Meanwhile, the entire motherboard [business](http://www.tomshardware.com/reviews/intel-uefi-firmware,2486-8.html) is pretty much supporting UEFI on paper. We found only a few exceptions that were mostly aimed at utilizing UEFI for improving visual appearances, unfortunately. Even though UEFI features are more interesting for system builders, the new standard also represents the only way to properly handle hard drives that exceed a total capacity of 2TB.

We failed in trying to get a 4TB RAID 0 array (2 x 2TB drives) to work on an Intel DP55KG, using its on-board Matrix RAID solution. It currently limits bootable arrays to a maximum of 2TB. Intel’s self-imposed limitations aside, we were fortunately successful in creating a 3TB system drive in Windows 7 with UEFI enabled using an external storage box by LaCie. The box employs RAID internally, and hence is a perfect showcase of what will happen once larger hard drives become available.

Finally, we can’t help but encourage everyone to keep an eye on the platform vendors and encourage them to bring their UEFI implementations up to speed. It feels wrong to purchase a brand new P55 motherboard today not knowing if it will be able to support future hard drive capacities. Keep in mind that this is all about booting from 2TB+ drives. If your boot drive is less than 2TB, you can still install virtually unlimited capacity in secondary system drives.

http://www.tomshardware.com/reviews/intel-uefi-firmware,2486.html