

Handreichung  
zur Installation von Windows 7-10  
in Schulen

# BIOS und UEFI

## Einführung und Unterschiede

Derzeit erfolgt eine Umstellung bei der Firmware der PC von BIOS auf UEFI. Während es bei den meisten Desktop PCs möglich ist, diesen im BIOS-kompatiblen Modus zu betreiben, ist dies bei einigen Windows-Tablets nicht mehr der Fall, so dass es sinnvoll oder notwendig ist – sofern man in die Installationsprozesse einsteigen möchte – sich mit diesem Thema zu beschäftigen.

### INHALT

BIOS .....	3
UEFI .....	4
Aufbau von Festplatten .....	6
Der GPT-Partitionsstil .....	7
Beispiel zum Einrichten einer MBR-Festplatte .....	10
Beispiel zum Einrichten einer GPT-Festplatte .....	11
UEFI oder BIOS.....	12
Empfehlungen .....	12

## IMPRESSUM

Akademie für Lehrerfortbildung und Personalführung, Dillingen (<http://alp.dillingen.de>)

Die Handreichung wurde im Rahmen des Projektes SCHULNETZ von Systembetreuern und IT-Multiplikatoren erarbeitet. Sie ist unter der Adresse <http://alp.dillingen.de/schulnetz/materialien> abrufbar.

Dokumentation: Georg Schlagbauer, Akademie Dillingen  
Christian Maushart, Landratsamt Dillingen  
E-Mail: [schlagbauer@alp.dillingen.de](mailto:schlagbauer@alp.dillingen.de)  
Stand: Juni 2018



## BIOS

### BIOS: BASIC INPUT OUTPUT SYSTEM

Das BIOS ist die klassische Firmware bei x86 PCs. Es stammt im Wesentlichen aus dem Jahr 1980. Nach dem Einschalten des PC führt das BIOS eine Hardwareerkennung durch (insbesondere Grafik, Netzwerk, Festplatten). Es legt fest, von welcher Festplatte gebootet werden soll und startet den Bootloader von dieser Festplatte.

Konkreter gesagt, ist jedoch das BIOS gar nicht in der Lage, einen Bootloader zu starten, bzw. das was wir üblicherweise unter einem Bootloader verstehen (z. B. NTLDR bei Windows XP, BootMGR bei Windows ab Vista, oder GRUB unter Linux).

Das BIOS kennt nur Festplatten, aber weder Partitionen noch Dateisysteme. Es würde den Bootloader auf der Festplatte gar nicht finden. Das einzige, was das BIOS kann, ist, auf die Festplatte zuzugreifen und diese der Reihe nach auszulesen. So greift es auf den ersten Block (512 Byte) der Festplatte (Master Boot Record, MBR) zu und liest den Inhalt. Wenn am Ende des MBR eine entsprechende Signatur (55AA) steht, wird der Inhalt in den Prozessor geladen und als Boot-Code ausgeführt.

512 Byte an Code sind nicht allzu viel. Außerdem enthält der MBR auch noch die Partitionstabelle (64 Byte) und die schon erwähnte Signatur (2 Byte), so dass als ausführbarer Code 446 Byte zur Verfügung stehen.

Bei Windows-Systemen muss eine der 4 möglichen Partitionen mit einem speziellen Flag versehen und damit als „aktive“ Partition markiert sein. Der Boot-Code im MBR liest die physikalische Adresse dieser Partition, springt dorthin und führt den Code weiter aus. Erst wenn auch dieser zweite Teil des Boot-Codes ausgeführt ist, ist das System so weit, dass es das Dateisystem interpretieren und den Bootmanager (z. B. das Programm bootmgr) laden und starten kann.

Andere Systeme (z. B. GRUB) nutzen einen (aus historischen Gründen) freien Platz zwischen MBR und erster Partition und legen dort einen weiteren Teil des Boot-Codes ab, der dann in der Lage sein soll, das Dateisystem zu lesen, auf dem sich der eigentliche Bootmanager befindet. Allerdings ist die Größe dieses freien Platzes nirgends definiert. Zudem nutzen auch andere Programme diesen freien Platz, um besonders zu schützenden Informationen (z. B. Lizenzinformationen) dort abzulegen.

Dieser Bootprozess funktioniert, ist aber nicht sehr flexibel.

Der Boot-Code muss sich an einer bestimmten Stelle auf der Festplatte (z. B. im MBR bzw. zu Beginn der aktiven Partition) befinden. Der Boot-Code ist auch nicht als Datei im Dateisystem erkennbar. Wenn Festplatten auf Dateiebene kopiert werden, ist der Boot-Code nicht mehr vorhanden.

Der BIOS-Bootmanager kennt nur Festplatten oder Disketten (später kamen noch CD, USB und PXE dazu) von denen gebootet werden soll. Eine weitere Differenzierung ist nicht möglich, z. B. kann man im BIOS nicht festlegen, dass von der ersten Festplatte Windows oder Linux gebootet werden kann.

Jedes Betriebssystem installiert seinen eigenen Boot-Code und richtet seinen eigenen Boot-Manager ein. Die Boot-Codes der unterschiedlichen Betriebssysteme überschrieben sich gegenseitig und arbeiteten eher gegeneinander als zusammen.



## UEFI

### UEFI: UNIFIED EXTENSIBLE FIRMWARE INTERFACE

Das UEFI kann deutlich mehr als das BIOS. Es bietet zusätzliche Möglichkeiten, die Hardware eines PC anzusprechen und auch erweiterte Bootmöglichkeiten. UEFI kann die Partitionsstruktur einer Festplatte interpretieren und auch FAT-Dateisysteme lesen. Damit ist es nicht mehr nötig, einen Boot-Code auf der Festplatte abzulegen, der an eine physikalische Adresse gebunden ist und nirgends im Dateisystem sichtbar ist.

### CSM-KOMPATIBILITÄTSMODUS

Um zumindest in einer Übergangszeit mit dem klassischen BIOS-Startsystem klar zu kommen, bieten die meisten Motherboard-Hersteller in ihrer UEFI-Firmware auch einen „BIOS-Kompatibilitäts-Modus“ (Compatibility Support Module, CSM) an, bei dem sich UEFI so verhält, wie das klassische BIOS, also den MBR-Bootloader startet und die weitere Initialisierung diesem Bootloader überlässt. Die Festplatte muss dazu im MBR-Format eingerichtet sein.

Dieser Boot-Modus wird als BIOS-Modus, BIOS-Kompatibilitätsmodus, CSM-Modus oder als Legacy-Modus angesprochen.

### NATIVES UEFI

Das native UEFI setzt üblicherweise voraus, dass die Festplatte, von der das Betriebssystem bootet, als GPT-formatierte Festplatte (GUID-Partitionstabelle) eingerichtet ist. Dies ist ein Grund, warum man zwischen CSM-Modus und nativem UEFI nicht so leicht wechseln kann, ohne die Festplatte neu zu einrichten und das Betriebssystem neu zu installieren.

Die UEFI-Firmware ist in der Lage, die Partitionstabelle einer mit GPT-formatierten Festplatte zu interpretieren. UEFI kann auch die Daten aus einer Partition mit einem FAT-Dateisystem (FAT12, FAT16, FAT32) lesen. UEFI definiert auch ein Format, wie ein Bootloader geschrieben werden muss, der von UEFI ausgeführt werden kann.

UEFI ist natürlich auch in der Lage, die Partitionstabelle einer MBR-formatierten Festplatte zu lesen. Dies wird z. B. beim Booten vom USB-Stick benötigt (UEFI-Fallback-Modus).

UEFI kann also

- Die Partitionstabelle einer GPT- oder MBR-Festplatte lesen
- Die Daten aus einem FAT-Dateisystem lesen
- Programmcode (Bootloader) in einem bestimmten Format ausführen

Damit können Betriebssysteme ihren Bootloader in einem definierten Format in einer ganz normalen Partition ablegen. UEFI definiert eine solche Partition als „EFI System Partition“ (ESP).

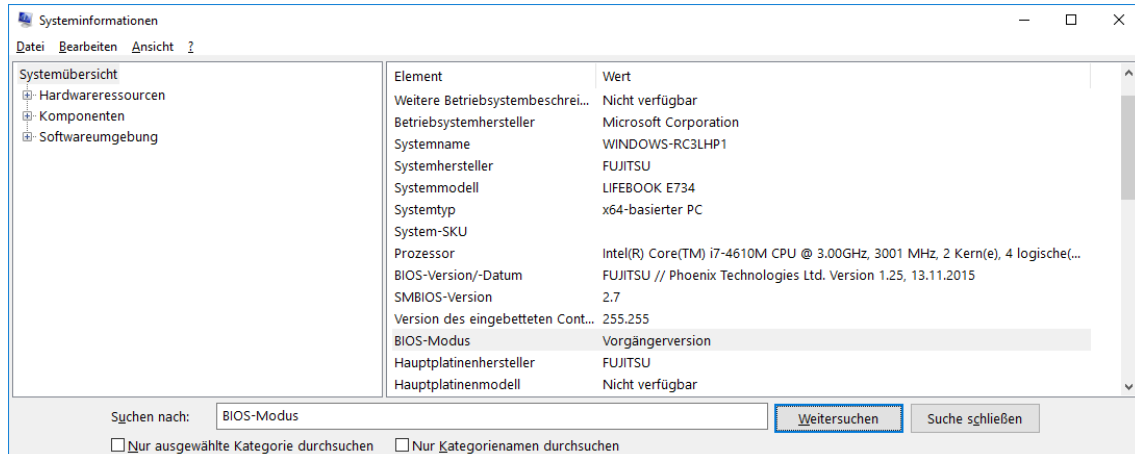


## UEFI-BOOT-MODI

UEFI kennt verschiedene Boot-Modi:

### BIOS-KOMPATIBILITÄTS-MODUS

Unterstützt die UEFI-Firmware CSM, kann der Computer im BIOS-Boot-Modus installiert werden. Unter Windows wird dies in den Systeminformationen (msinfo32) angezeigt (BIOS-Modus: Vorgängerversion).



### VOLLSTÄNDIGER UEFI-BOOT-MODUS

Vollständige UEFI-Einträge sind gedacht, um installierte Betriebssysteme zu starten. Sie referieren einen Eintrag in der ESP (EFI-Systempartition) in der Form `\EFI\MICROSOFT\BOOT\BOOTMGFW.EFI` oder `\EFI\UBUNTU\GRUBX64.EFI`.

Die Vorstellungen von UEFI sind, dass alle Betriebssysteme einen eigenen Bootloader in einer EFI-System-Partition einrichten, der nur auf das eigene Betriebssystem verweist und davon absieht, auch andere Betriebssysteme zu booten.

### UEFI-FALLBACK BOOT-MODUS

Der UEFI-Fallback-Modus ist gedacht, um das System von einem Wechselmedium (z. B. USB-Stick) zu starten. UEFI durchsucht das Wechselmedium nach FAT32-Partitionen und sucht in diesen Partitionen nach einem Eintrag `\EFI\BOOT\BOOTx64.EFI` (bei x86-64-Architekturen). Der erste gefundene Eintrag dieser Art wird ausgewertet.

Wenn ein entsprechender Eintrag gefunden wird, wird dieser im UEFI-Bootmenü zur Auswahl angeboten.

Der Fallback-Modus ist gedacht, um für Live- oder Installations-Systeme von Wechselmedien zu starten. Diese Wechselmedien enthalten eine FAT-formatierte Partition. In dieser Partition gibt es die Datei `\EFI\BOOT\BOOTx64.EFI`.



## DER UEFI-BOOTMANAGER

Der UEFI-Bootmanager könnte Ordnung beim Booten des PCs in das System bringen – wenn sich die Motherboard-Hersteller an die Spezifikationen halten würden und auch Werkzeuge zur Pflege des Boot-Managers anbieten würden.

Der UEFI-Bootmanager kann Betriebssysteme auf einer Festplatte oder auf einem Bootmedium identifizieren. Es gibt auch Werkzeuge mit denen man den UEFI-Bootmanager von einem Betriebssystem aus bearbeiten kann. Unter Windows z. B. mit dem Programm „EasyUEFI“, unter Linux mit „efibootmgr“.

## INSTALLATION VON BETRIEBSSYSTEMEN

Ein Wechselmedium (z. B. USB-Stick) kann so eingerichtet sein, dass dieses sowohl im BIOS-Modus als auch im UEFI-Modus booten kann. Falls das UEFI-Setup entsprechend eingestellt ist, werden im Bootmenü für ein Installationsmedium zwei Einträge angezeigt, z. B.:

UEFI: JetFlashTranscend 16 GB, Partition 1

JetFlashTranscend 16 GB

...

Startet man das Installationsmedium im UEFI-Modus, wird es versuchen, das Betriebssystem ebenfalls im UEFI-Modus zu installieren. Startet man das Installationsmedium im BIOS-kompatiblen Modus, wird es versuchen, einen MBR-Bootloader zu installieren.

## SECURE BOOT

Secure Boot ist eine Möglichkeit, das Starten von nicht signierten Boot-Loadern zu verhindern. Im UEFI sind öffentliche Schlüssel von vertrauenswürdigen Anbietern hinterlegt und nur wenn ein Bootloader mit dem zugehörigen privaten Schlüssel signiert wurde, lässt er sich starten.

Auf x86-Computern kann Secure Boot ausgeschaltet werden. Zusätzlich sollte es (nach UEFI-Spezifikation) möglich sein, im UEFI-Setup Schlüssel hinzuzufügen oder zu entfernen.

## AUFBAU VON FESTPLATTEN

- Bei einer BIOS-kompatiblen Installation muss die Festplatte im MBR-Partitionsstil eingerichtet sein.
- Bei einer UEFI-Installation muss die Festplatte im GPT-Partitionsstil eingerichtet sein.

## DER MBR-PARTITIONSSSTIL

Der MBR-Partitionsstil ist der klassische Standard, wie eine Festplatte aufgebaut ist. Er wurde Anfang der 1980-Jahre definiert und seitdem einige Male geringfügig angepasst, um auch größere Festplatten (bis 2 Terabyte) ansprechen zu können.



## GRUNDSÄTZLICHER AUFBAU VON MBR-FESTPLATTEN

- MBR (Master Boot Record) mit 512 Byte
- maximal 4 primäre Partitionen

Innerhalb des MBR sind 4 x 16 Byte reserviert, um max. 4 Partitionen adressieren zu können (Primäre Partitionen). Diese 16 Byte sind allerdings noch aufgeteilt, so dass für die eigentliche LBA-Adresse (Logical Block Addressing) nur 4 Byte zur Verfügung stehen, so dass die maximal adressierbare Größe einer Partition bei 2 Terabyte liegt ( $2^{32}$  Blöcke x 512 Byte/Block = 2 Terabyte).

Eine der primären Partitionen kann eine „aktive“ Partition sein. Der Computer startet von dieser aktiven Partition (falls der MBR nicht mit einem anderen Bootloader überschrieben wurde).

Eine der primären Partitionen kann zu einer erweiterten Partition gemacht werden, innerhalb der mehrere logische Laufwerke definiert werden können. Die logischen Laufwerke bilden eine verkettete Liste.

Zwischen MBR und der ersten Partition ist aus historischen Gründen etwas freier Platz. Verschiedene Programme legen dort Informationen ab, die besonders geschützt sein sollen.

## SYSTEMPARTITION UND STARTPARTITION EINER WINDOWS-INSTALLATION (MBR)

Bei der Installation von Windows werden standardmäßig zwei Partitionen angelegt:

- Die **Systempartition** (je nach Windows-Version 100-500 MB) ist die erste Partition der Festplatte und ist als aktive Partition gekennzeichnet. Sie enthält den Bootmanager.
- Die **Startpartition** oder Windows-Partition enthält die Windows-Betriebssystemdateien, insbesondere das Windows-Verzeichnis.

<b>Datenträger 0</b> Basis 476,94 GB Online	<b>System-reserviert</b> 500 MB NTFS Fehlerfrei (System, Aktiv, Primäre Partition)	<b>(C:)</b> 476,45 GB NTFS Fehlerfrei (Startpartition, Auslagerungsdatei, Absturzabbild, Primäre Partition)
--	--	---

Abbildung: Partitionierung der Festplatte, aus dem laufenden Windows-Betriebssystem heraus gesehen. Die Systempartition hat keinen Laufwerksbuchstaben und ist deshalb für einen Benutzer nicht erreichbar.

Die Trennung von Systempartition und Startpartition ist nicht zwingend notwendig, hat aber den Vorteil, dass der Windows-Bootmanager getrennt vom Betriebssystem ist und nicht versehentlich verändert werden kann. Einige Imaging-Programme kommen jedoch besser zurecht, wenn sie nur eine Partition sichern müssen.

## DER GPT-PARTITIONSTIL

Der GPT-Partitionsstil kann (nach heutigen Maßstäben) beliebig große Festplatten und beliebig viele Partitionen ansprechen.



---

## GRUNDSÄTZLICHER AUFBAU VON GPT-FESTPLATTEN

- MBR (Master Boot Record) mit 512 Byte als "Protective MBR"
- Primäre GUID-Partitionstabelle
- max. 128 Partitionen
- Backup GUID-Partitionstabelle

Der MBR ist so vorbelegt, dass ältere Werkzeuge, die mit GPT noch nicht umgehen können, die Information erhalten, dass die gesamte Festplatte belegt ist (Protective MBR).

Die einzelnen Partitionen sind durch den Partitionstyp (16 Byte Hexadezimalzahl) in ihren Funktionen unterscheidbar (z. B. EFI-Systempartition, MSR-Partition, Windows-Recovery-Partition, Windows-Partition, Linux-Partition, etc.)

---

## EFI-SYSTEMPARTITION (ESP)

Die EFI-Systempartition enthält den Boot-Manager. Sie sollte keine weiteren Daten enthalten.

- Mindestgröße: 100 MB
- Dateisystem: FAT32
- Typ: c12a7328-f81f-11d2-ba4b-00a0c93ec93b

---

## MICROSOFT RESERVED PARTITION (MSR-PARTITION)

Der Zweck dieser Partition ist noch nicht ganz klar. Microsoft kann die Partition für zukünftige Zwecke verwenden. In etwa entspricht diese Partition dem freien Platz, der bisher bei MBR-Partitionen zur Verfügung stand.

- Größe (bei Windows 10): 16 MB
- kein Dateisystem
- Typ: e3c9e316-0b5c-4db8-817d-f92df00215ae

---

## WINDOWS-PARTITION

Diese Partition enthält das Windows-Betriebssystem

- Mindestgröße: 20 GB
- Dateisystem: NTFS
- Typ: ebd0a0a2-b9e5-4433-87c0-68b6b72699c7





## WIEDERHERSTELLUNGSPARTITION

Enthält die Datei winre.wim (Windows Recovery).

- Standardgröße bei Windows 10: 500 MB
- Dateisystem: NTFS
- Typ: de94bba4-06d1-4d40-a16a-bfd50179d6ac

## SYSTEMPARTITION UND STARTPARTITION EINER WINDOWS-INSTALLATION (GPT)

Bei der folgenden Windows-Installation wurde auf die Wiederherstellungspartition verzichtet.

```
DISKPART> list disk
```

Datenträger ###	Status	Größe	Frei	Dyn	GPT
Datenträger 0	Online	476 GB	0 B		*
Datenträger 1	Online	465 GB	0 B		

Diskpart zeigt an, dass der Datenträger 0 als GPT-Datenträger eingerichtet ist.

```
DISKPART> list partition
```

Partition ###	Typ	Größe	Offset
Partition 1	System	100 MB	1024 KB
Partition 2	Reserviert	16 MB	101 MB
Partition 3	Primär	476 GB	117 MB

Datenträger 0	Partitionen									
Basis 476,92 GB Online	<table border="1"> <thead> <tr> <th>Partition</th> <th>Größe</th> <th>Typ</th> </tr> </thead> <tbody> <tr> <td>100 MB</td> <td>100 MB</td> <td>Fehlerfrei (EFI-Systempartition)</td> </tr> <tr> <td>476,83 GB NTFS</td> <td>476,83 GB</td> <td>Windows (C:) Fehlerfrei (Startpartition, Auslagerungsdatei, Absturzabbild, Primäre Partition)</td> </tr> </tbody> </table>	Partition	Größe	Typ	100 MB	100 MB	Fehlerfrei (EFI-Systempartition)	476,83 GB NTFS	476,83 GB	Windows (C:) Fehlerfrei (Startpartition, Auslagerungsdatei, Absturzabbild, Primäre Partition)
Partition	Größe	Typ								
100 MB	100 MB	Fehlerfrei (EFI-Systempartition)								
476,83 GB NTFS	476,83 GB	Windows (C:) Fehlerfrei (Startpartition, Auslagerungsdatei, Absturzabbild, Primäre Partition)								

In der grafischen Ansicht wird die MSR-Partition nicht angezeigt.



**BEISPIEL ZUM EINRICHTEN EINER MBR-FESTPLATTE**

Im folgenden Beispiel wird eine Festplatte (disk 0) komplett gelöscht und neu eingerichtet:

- Partition 1 500 MB Systempartition
- Partition 2 Rest Windows-Startpartition

```
diskpart
DISKPART> list disk
DISKPART> select disk 0
DISKPART> clean
DISKPART> create partition primary size=500
DISKPART> format fs=ntfs quick label="Hide"
DISKPART> active
DISKPART> assign letter=s
DISKPART> create partition primary
DISKPART> format fs=ntfs quick label="Windows 10"
DISKPART> assign letter=w
DISKPART> exit
```

Die beiden „assign“-Befehle und die zugewiesenen Laufwerksbuchstaben haben nur eine temporäre Bedeutung, damit man sofort nach dem Einrichten (ohne Neustart) auf die Laufwerke zugreifen kann.



**BEISPIEL ZUM EINRICHTEN EINER GPT-FESTPLATTE**

Im folgenden Beispiel wird eine Festplatte (disk 0) komplett gelöscht und neu eingerichtet:

- Partition 1 100 MB Systempartition
- Partition 2 16 MB MSR-Partition (Microsoft Reserved)
- Partition 3 Rest Windows-Partition
- Partition 4 500 MB Recovery-Partition

```
diskpart
DISKPART> list disk
DISKPART> select disk 0
DISKPART> clean
DISKPART> convert gpt
DISKPART> create partition efi size=100
DISKPART> format fs=fat32 quick label="System"
DISKPART> assign letter=s
DISKPART> create partition msr size=16
DISKPART> create partition primary
DISKPART> shrink minimum=500
DISKPART> format fs=ntfs quick label="Windows"
DISKPART> assign letter=w
DISKPART> create partition primary
DISKPART> format fs=ntfs quick label="Recovery"
DISKPART> set id="de94bba4-06d1-4d40-a16a-bfd50179d6ac"
DISKPART> gpt attributes=0x8000000000000001
DISKPART> exit
```

Auf die Recovery-Partition kann beim Klonen auch verzichtet werden, so dass sich die diskpart-Befehle etwas reduzieren.



## UEFI ODER BIOS

In bestimmten Fällen hat man keine Wahl, ob man UEFI oder den BIOS-kompatiblen CSM-Modus verwendet. Dies ist z. B. der Fall, wenn UEFI keinen CSM-Modus mehr unterstützt, wie dies z. B. bei Windows-Tablets der Fall ist. Wenn man eine große Festplatte hat (> 2 Terabyte), von der man booten möchte, muss man ebenso UEFI verwenden. Sollen mehrere Betriebssysteme installiert werden, bringt UEFI zumindest Vorteile.

Nachteilig an UEFI ist, dass die Bedienoberfläche des Bootmanagers innerhalb der UEFI-Firmware bei vielen Motherboard-Herstellern noch nicht ganz ausgereift scheint. Ein weiterer Nachteil kann sein, dass die Architektur von UEFI und Betriebssystem zusammen passen müssen. Auf einem PC muss man zwischen 32- und 64-bit-Systemen unterscheiden. Ein 64-bit-UEFI kann nur ein 64-bit Betriebssystem booten. Dies gilt auch für Windows- oder Linux-Live-Systeme. Dem BIOS-Kompatibilitätsmodus ist dies egal. Dieser bootet alles.

Das UEFI-Startmedium muss mit FAT32 partitioniert sein oder zumindest eine FAT32-Partition aufweisen, in der die Boot-Dateien liegen. Dem BIOS-Kompatibilitätsmodus ist dies wiederum egal. Dieser kümmert sich nicht um ein Dateisystem.

## EMPFEHLUNGEN

Es ist natürlich schwierig, Systembetreuern, die primär nur ihren eigenen Erfahrungen glauben, Empfehlungen zu geben. Aber trotzdem:

- Installieren Sie auf **einem** Computer nur **ein** Betriebssystem. Ansonsten sollten Sie sich mit Bootmanagern wirklich auskennen. Wenn Sie mehrere Betriebssysteme brauchen, sind virtuelle Maschinen die bessere Wahl.
- Entscheiden Sie sich zwischen BIOS-Kompatibilität und UEFI. Die vom UEFI-Setup angebotene Einstellung „UEFI and BIOS“ bzw. „UEFI and Legacy“ wird zum Chaos führen.
- Wenn Sie die freie Wahl haben zwischen BIOS-Kompatibilität und UEFI, wählen Sie BIOS. Auch wenn der Bootvorgang etwas chaotisch erscheint und angeblich etwas länger dauert, so funktioniert er doch.

