



Installation von openSUSE auf ein externes Laufwerk

Es wird die Vorgehensweise bei der Installation von openSUSE auf einem externen Laufwerk d.h. einer USB- oder Firewire-Festplatte beschrieben.

Im **Kapitel 2 „Vorbereitungen“** wird ein Vorschlag für die Partitionierung der externen Platte gemacht und eine eventuell erforderliche Anpassung der internen Platte erläutert.

Eine Besonderheit betrifft bei Firewire-Anschlüssen die Initial RAM Disk (*initrd*), einem virtuellen Laufwerk, über das dem Installationsprozess Informationen über die zu installierenden Module bereitgestellt werden. Normalerweise gibt es auf der *initrd* keine Hinweise auf Firewire-Module. Daher muss die *initrd* nach der Installation von openSUSE entsprechend angepasst werden. Dies wird in **Kapitel 4.1 „Anpassen der Initialen RAM-Disk“** beschrieben.

Anmerkung:

Dies gilt auch für USB-Laufwerke bei früheren SuSE-Versionen. Ab Version 10.2 sind die erforderlichen USB-Module in der *initrd* bereits enthalten.

Ein weiteres Problem stellt der Boot von openSUSE dar. Die verschiedenen Boot-Möglichkeiten wie Boot über den MBR der internen Platte, Boot über das BIOS-Bootmenü, Boot über Floppy oder CD und Boot über das Windows-Startmenü werden in **Kapitel 4.3 „Bootmöglichkeiten“** beschrieben.

Inhaltsverzeichnis

1 Motivation.....	2
2 Vorbereitungen.....	2
2.1 Einrichten der externen Platte.....	2
2.2 Einrichten eines USB-Sticks.....	3
2.3 Einrichten einer boot-Partition für Linux auf der internen Platte.....	3
3 Besonderheiten während der Installation.....	4
3.1 Mountpunkte setzen.....	4
3.2 Einrichten des Bootloaders.....	4
3.3 Erster Boot.....	4
4 Aktivitäten nach der Installation.....	4
4.1 Anpassen der „Initialen RAM Disk (initrd)“.....	4
4.1.1 USB ab openSUSE 10.2.....	4
4.1.2 Frühere Versionen.....	4
4.1.3 IEEE 1394.....	5
4.2 Speicherort des Bootloaders.....	5
4.3 Bootmöglichkeiten.....	5
4.3.1 Boot von GRUB-Diskette.....	5
4.3.2 Boot von GRUB-CD.....	5
4.3.3 Boot von CD mit Bootmanager isolinux.....	7
4.3.4 GRUB im Startmenü von Windows.....	8
4.3.4.1 Starten von Linux mit grldr über das Windows XP Bootmenü.....	8
4.3.4.2 Starten von Linux mit grldr über das Windows Vista Bootmenü.....	8
4.3.4.3 Anmerkung zu openSUSE 11.0.....	9
4.3.4.4 Optionale Schritte.....	9
4.3.5 Wenn BIOS die externe Platte nicht erkennt.....	10
4.3.6 GRUB im MBR der internen Platte bei eigener Bootpartition für Linux auf der internen Platte.....	10

1 Motivation

Externe Laufwerke, angeschlossen über *USB* oder *firewire (IEEE 1394)*, sind inzwischen sehr kostengünstig. Damit eröffnet sich eine einfache Möglichkeit, die Festplattenkapazität eines Notebooks oder PCs zu vergrößern. Man kann relativ problemlos zusätzlich Linux z. B. openSUSE auf einer externen Platte installieren. Windows bleibt dann auf der internen Festplatte unangetastet erhalten.

Bezeichnungen:

Im Folgenden wird die im Rechner eingebaute Festplatte, die das Laufwerk C:\ enthält, als **interne Platte** bezeichnet.

Die über USB oder IEEE 1394 angeschlossene Festplatte wird als **externe Platte** bezeichnet.

2 Vorbereitungen

2.1 Einrichten der externen Platte

Auf der externen Platte sollten angelegt werden:

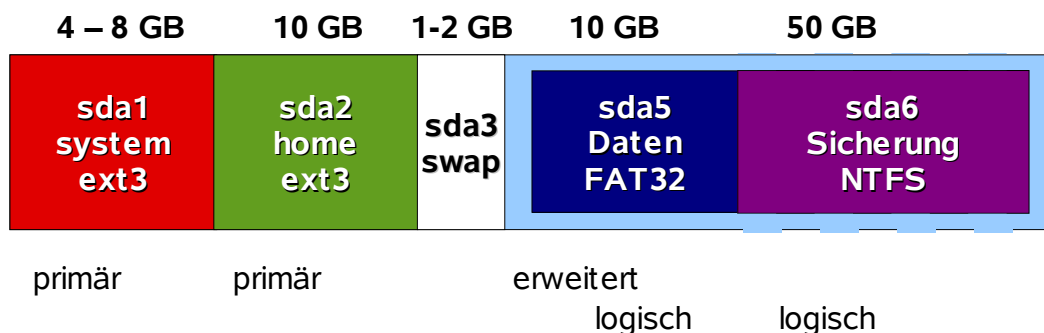
- eine primäre ext3-Partition mit 4 - 8 GB für das System mit *Mountpoint /*
- eine primäre ext3-Partition mit etwa 10 GB für den home-Bereich mit *Mountpoint / home*
- eine primäre swap-Partition mit 1 - 2 GB.

Dann ist mit dem restlichen Platz eine erweiterte Partition anzulegen.

Darin ist dann Platz für:

- eine logische FAT32-Partition mit etwa 10 GB mit *Mountpoint /windows/X* für den gemeinsamen Datenzugriff von windows und Linux, wobei X durch den gewünschten Laufwerksbuchstaben zu ersetzen ist und
- eine logische Partition für z. B. Sicherung wichtiger Daten der internen Platte, z. B. mit NTFS formatiert.

Externe Platte mit z.B. 80 GB:



2.2 Einrichten eines USB-Sticks

Im Prinzip kann man anstelle einer externen Festplatte auch einen USB-Stick zur Installation von openSUSE verwenden. Dazu ist eine Kapazität von 8 GB erforderlich.

Auf einem Corsair Flash Survivor 8 GB mit nominellen Lese- und Schreibzugriffszeiten von 20 MB/s dauerte die Installation von openSUSE 11.0 etwa 2 Stunden. Die Reaktionszeiten beim Arbeiten mit diesem System sind z.T. schleppend. Verlagert man den swap-Bereich auf die Festplatte, verbessert sich das Verhalten.

Da auch ein schneller USB-Stick erheblich langsamere Zugriffe als eine Festplatte aufweist, stellt sich die Frage, zu welchem Zweck man openSUSE auf einem USB-Stick benötigt. Da es sich nicht um ein Life-System handelt, ist dieses openSUSE-System ohne aufwendige Anpassungen nur auf dem Rechner lauffähig, auf dem es erstellt worden ist.

Das Argument, dass es sich bei einem USB-Stick um ein ideales Zubehör für mobile Notebooks handelt, stimmt so auch nicht mehr, seit es die externen 2,5“-USB-Laufwerke gibt.

Anmerkung:

Normalerweise kann ein USB-Laufwerk, also auch ein USB-Stick, über das BIOS gebootet werden. Sollte der Boot vom USB-Stick nicht funktionieren, kann über `cdisk` das Bootflag für die Root-Partition auf dem USB-Stick gesetzt werden.

2.3 Einrichten einer boot-Partition für Linux auf der internen Platte

Wenn das BIOS die externe Platte beim Boot nicht erkennt oder wenn der *Bootloader GRUB* im *MBR* der internen Platte abgelegt werden soll, dann muss auf der internen Platte eine kleine Linux-Partition von etwa 30 MB angelegt und z. B. mit ext3 formatiert werden.

Auf diese Partition kann dann bei der Installation der *Mountpoint /boot* gelegt werden und damit eine eigene Boot-Partition angelegt werden.

Hat man dies bei der Installation von Linux versäumt anzugeben, kann man auch nachträglich das Verzeichnis */boot* von der externen Platte hierher kopieren.

3 Besonderheiten während der Installation

3.1 Mountpunkte setzen

Bei der Installation von openSUSE muss man im Bildschirm „Installationseinstellungen“ auf den Reiter „Experten“ gehen und nach Abschluss der „Systemanalyse“ wählen:

Ändern > Partitionierung > Benutzerdefiniertes Partitions-Setup erstellen > Benutzerdefinierte Partitionierung für Experten

Hier lassen sich dann ganz individuell die Mounpunkte / für root, /home für den Home-Bereich, swap sowie /windows/X für den Datenbereich setzen und falls erforderlich auch /boot für den Boot-Bereich.

3.2 Einrichten des Bootloaders

Der Bootloader muss auf der externen Platte gespeichert werden (siehe dazu Kap. 4.2). Dazu im Bildschirm „Installationseinstellungen“

Ändern > Systemstart > Bootloadereinstellungen > Bootloader Installation
wählen und

Boot from Root Partition
ankreuzen **und nur das ankreuzen!!**

3.3 Erster Boot

Bei der Installation von openSUSE wird ein „erster Boot“ notwendig. Dazu den Rechner über das BIOS-Boot-Menü von USB bzw IEEE 1394 booten.

Wenn dies nicht möglich ist, muss nochmals von der Installations-DVD gebootet werden: Also erneut von der Linux-DVD booten, wieder Installation anklicken usw. bis zum Bildschirm "Installationsmodus".

"Weitere Optionen" wählen

"Installiertes System starten" wählen

Bei openSUSE 11.0 fehlt dieses Kästchen. Man muss <Abbrechen> wählen und dann <Installation/System starten> und <Installiertes System starten> aufrufen.

Danach folgt die abschließende Konfiguration von Linux.

4 Aktivitäten nach der Installation

4.1 Anpassen der „Initialen RAM Disk (initrd)“

Damit Linux das externe Laufwerk verwenden kann, muss die *initrd* angepasst werden, d. h. es müssen die Module für die externe Platte eingetragen werden. Welche das sind, kann man mit Hilfe des Kommandos *lsmod* feststellen.

4.1.1 USB ab openSUSE 10.2

Spätestens ab openSUSE 10.2 enthält die *initrd* bereits die erforderlichen USB-Module für die externe Platte. Es sind

```
ehci_hcd usb_storage
```

Die gegenwärtige *initrd* lautet z. B.:

```
INITRD_MODULES="processor thermal piix ahci ata_piix fan jbd ext3  
ehci_hcd usb_storage edd"
```

4.1.2 Frühere Versionen

Bei früheren Versionen müssen z. B. die folgenden USB-Module hinzugefügt werden:

```
usbcore usb_storage uhci_hcd usbhid ehci_hcd
```

Wichtig ist die Reihenfolge:

die USB-Module müssen **nach** ext3 (bzw. reiserfs) eingefügt werden!

Das Eintragen lässt sich über *YaST /System /Editor für sysconfig-Dateien* bewerkstelligen.

4.1.3 IEEE 1394

Bei IEEE 1394 müssen z. B. folgende Module hinzugefügt werden:

```
ieee1394 sbp2 ohci1394
```

Das Eintragen lässt sich über *YaST /System /Editor für sysconfig-Dateien* bewerkstelligen.

4.2 Speicherort des Bootloaders

Das einzig größere Problem ist dann die Frage, wie man Linux auf der externen Platte booten kann. Die einfachste Möglichkeit, die Installation des *Bootloaders GRUB* im *Master Boot Record (MBR)* der internen Festplatte scheidet aus, weil dann bei nicht angeschlossenem externen Laufwerk nichts mehr - auch nicht Windows – gebootet werden kann.

GRUB/stage1 im MBR startet GRUB/stage2, das in der Linux-Partition unter `/boot/grub/stage2` liegt. GRUB/stage2 lädt als erstes die Datei `/boot/grub/menu.lst` und zeigt sie an.

Wenn also die USB-Platte nicht eingeschaltet ist, kann auf stage2 und damit auf die `menu.lst` auch nicht zugegriffen werden und das Bootmenü kann nicht geladen werden.

Wenn die externe Festplatte nicht angeschlossen ist, kann GRUB diese Datei naturgemäß nicht finden und läuft auf einen Fehler.

GRUB muss daher auf der externen Platte gespeichert werden. Unter

YaST / System / Konfiguration des Bootloaders / Bootloader-Installation

sollte man dazu „**Boot from Root Partition**“ ankreuzen.

Dann wird Windows wie bisher automatisch nach dem Einschalten gestartet.

4.3 Bootmöglichkeiten

Will man Linux auf der externen Platte booten, gibt es folgende Möglichkeiten:

- Boot von GRUB-Diskette
- Boot von GRUB-CD
- Boot von CD mit Bootmanager `isolinux`
- GRUB im Startmenü von Windows
- GRUB im MBR der internen Platte bei eigener Bootpartition für Linux auf der internen Platte

4.3.1 Boot von GRUB-Diskette

Eine Diskette für den Boot mit GRUB lässt sich mit root-Rechten in der Konsole wie folgt erzeugen:

```
/usr/bin/grub-install /dev/fd0
```

4.3.2 Boot von GRUB-CD

Dies wird in der openSUSE-Dokumentation in Kap. 14.5 beschrieben, die im folgenden zitiert wird.
file:///usr/share/doc/manual/opensuse-manual_de/manual/sec.grub.bootcd.html

Es werden root-Rechte benötigt.

- Erstellen eines Verzeichnisses */tmp/iso* und eines Unterverzeichnisses */tmp/iso/grub*
Dazu kann man den *konqueror* im *Systemverwaltermodus* benutzen oder in einer Konsole folgende Befehle eingeben:

```
cd /tmp
mkdir iso
mkdir -p iso/boot/grub
```

Kopieren von *stage2_eltorito* und *menu.lst* nach *iso/boot/grub*

und von *vmlinuz*, *initrd* und *message* nach *iso/boot/*

Dazu kann man den *konqueror* im *Systemverwaltermodus* benutzen oder in einer Konsole folgende Befehle eingeben:

```
cp /usr/lib/grub/stage2_eltorito iso/boot/grub/
(Achtung hier ist in der Originaldokumentation ein Fehler)
cp /boot/vmlinuz iso/boot/
cp /boot/initrd iso/boot/
cp /boot/message iso/boot/
cp /boot/grub/menu.lst iso/boot/grub/
```

- Anpassen der Pfadeinträge in *iso/boot/menu.lst* so, dass sie auf ein CD-ROM-Laufwerk verweisen. Hierfür in den Pfadnamen den Gerätenamen der Festplatten, die im Format (hd*) oder (sd*) aufgeführt sind, mit dem Gerätenamen des CD-ROM-Laufwerks, das mit (cd) angegeben wird, ersetzen.

Dazu kann man den Editor *kate* mit Rootrechten starten (*kdesu kate*).

```
default 0
timeout 8
gfxmenu (cd)/boot/message
title Linux
    kernel (cd)/boot/vmlinuz root=/dev/sda5 vga=794 \
    resume=/dev/sda1 splash=silent showopts
    initrd (cd)/boot/initrd
```

Dabei sind für *sda5* und *sda1* die Gerätebezeichnungen für die Linux-Partition und die swap-Partition einzugeben.

- Das ISO-Image kann mit *k3b* erzeugt und auf CD gebrannt werden:

k3b mit root-Rechten starten über
kdesu k3b

Neues Daten-CD-Projekt anklicken

Volume Name: z. B. GRUB_von_CD

Im Browser-Feld */tmp/iso/boot* suchen und ins Projektfeld ziehen.

Das Kästchen „Edit boot images ...“ neben „Rename Audio Files“ anklicken.

New

/tmp/iso/boot/grub/stage2_eltorito suchen und öffnen.

Emulationstyp: None

Show advanced options

Boot-info-table ankreuzen

ok

CD einlegen und

Burn

anklicken.

Wenn ein Fenster aufgeht
„Ein neues Medium wurde gefunden.“
„Nichts unternehmen“ anklicken.
ok

Über die Konsole lässt sich das ISO-Image mit dem folgenden Befehl erzeugen:

```
mkisofs -R -b boot/grub/stage2_eltorito -no-emul-boot \
-boot-load-size 4 -boot-info-table -o grub.iso iso
```

Es wird die Datei *grub.iso* erzeugt und unter */tmp* abgelegt.

Manchmal führt auch die folgende Variante zum Erfolg

```
mkisofs -R -b boot/grub/stage2_eltorito -no-emul-boot \  
-boot-load-size 4 -boot-info-table -o grub.iso /tmp/iso
```

Die so erstellte Datei *grub.iso* ist dann z. B. mit *k3b* als ISO-Image auf CD zu brennen.

4.3.3 Boot von CD mit Bootmanager isolinux

Es gibt Fälle, in denen die GRUB-CD nichtfunktioniert. Dann führt das Erstellen einer Boot-CD zum Erfolg, die auf dem Bootloader *isolinux* basiert. Die folgende Beschreibung basiert auf

http://de.opensuse.org/SDB:System-Boot-CD_erzeugen

Es werden Root-Rechte benötigt.

- Das Paket *syslinux* muss installiert werden.
- Erstellen eines Verzeichnisses */tmp/iso1*
Dazu kann man den *konqueror* im *Systemverwaltermodus* benutzen oder in einer Konsole folgende Befehle eingeben:

```
mkdir /tmp/iso1
```

- Kopieren von */usr/share/syslinux/isolinux.bin* und */boot/initrd* nach */tmp/iso1* und von */boot/vmlinuz* nach */tmp/iso1/linux*
Dazu kann man den *konqueror* im *Systemverwaltermodus* benutzen oder in einer Konsole folgende Befehle eingeben:

```
cp /usr/share//syslinux/isolinux.bin /tmp/iso1/  
cp /boot/vmlinuz /tmp/iso1/linux  
cp /boot/initrd /tmp/iso1
```

- Erstellen der Bootloader-Konfigurationsdatei */tmp/iso1/isolinux.cfg* mit *konqueror* und Eintragen folgender Zeilen mit einem Editor:

```
DEFAULT linux  
LABEL linux  
    KERNEL linux  
    APPEND initrd=initrd root=/dev/sda5 vga=794 \  
    splash=silent showopts
```

Dabei ist *sda5* durch die Bezeichnung der Linux-Partition zu ersetzen und der Wert für *vga* ist ggf anzupassen.

- Das ISO-Image kann mit *k3b* erzeugt und auf CD gebrannt werden:
k3b mit root-Rechten starten über
`kdesu k3b`

Neues Daten-CD-Projekt anklicken

Volume Name: z. B. *Boot_mit_Isolinux*

Im Browser-Feld */tmp/iso1* suchen und die Dateien ins Projektfeld ziehen.

Das Kästchen „Edit boot images ...“ neben „Rename Audio Files“ anklicken.

New

/tmp/iso1/isolinux.bin suchen und öffnen.

Emulationstyp: None

Show advanced options

Boot-info-table ankreuzen

Wenn ein Fenster aufgeht
„Ein neues Medium wurde gefunden.“
„Nichts unternehmen“ anklicken.
ok

ok
CD einlegen und
Burn
anklicken.

Über die Konsole lässt sich das ISO-Image mit dem folgenden Befehl erzeugen:

```
mkisofs -o /tmp/bootcd.iso -b isolinux.bin -c boot.cat \  
-no-emul-boot -boot-load-size 4 -boot-info-table /tmp/isol
```

Es wird die Datei *bootcd.iso* erzeugt und unter */tmp* abgelegt.

Die so erstellte Datei *bootcd.iso* ist dann z. B. mit *k3b* als ISO-Image auf CD zu brennen.

4.3.4 GRUB im Startmenü von Windows

Der GRUB-Aufruf lässt sich auch in das Startmenü von Windows integrieren. Dazu benötigt man das Paket *grub4dos*.

GRUB for DOS oder *GRUB4DOS* ist ein universeller Bootlader, der auf *GNU GRUB* basiert. Er kann direkt von DOS oder Linux gestartet werden, aber auch über den Windows Bootmanager oder *syslinux* oder *lilo*. Weiterhin kann er über den MBR oder von CD gestartet werden.

Die Hauptfunktion von GRUB ist in der Datei *grldr* enthalten. Die ersten 16 Sektoren von *grldr* enthalten den Bootlader, der zusätzlich in der Datei *grldr.mbr* abgelegt ist.

GRUB for DOS enthält man über

<https://sourceforge.net/projects/grub4dos>

4.3.4.1 Starten von Linux mit *grldr* über das Windows XP Bootmenü

Um über da XP Bootmenü Linux zu starten genügt es, in die *boot.ini* eine Zeile in der Form

```
C:\grldr="Linux über GRUB4DOS"
```

einzutragen und die Datei *grldr* nach *C:* zu kopieren.

Es wird dann im Windows Bootmenü eine neue Option angezeigt, mit der GRUB4DOS gestartet werden kann. GRUB4DOS durchsucht dann alle Partitionen nach der Datei *menu.lst*, welche das Startmenü von GRUB enthält.

4.3.4.2 Starten von Linux mit *grldr* über das Windows Vista Bootmenü

Bei Windows Vista gibt es die *boot.ini* nicht mehr. Statt dessen wurde die Binärdatei BCD (Boot Configuration Data Store) eingeführt, die sich nur noch über das Kommandozeilen-Tool *bcdedit.exe* bearbeiten lässt.

Es geht aber auch einfacher. Spendiert man Vista eine *boot.ini* unter *C:*, dann erinnert sich Windows an seine Vergangenheit und zeigt das in *boot.ini* konfigurierte Bootmenü an.

Weiterhin wird *grldr.mbr* unter *C:* benötigt, weil diese Datei klein genug ist, um vom Vista Bootmanager gestartet werden zu können. Der Vista Bootmanager kann nur Dateien starten, die kleiner als 64 MB sind.

Die *boot.ini* kann folgendermaßen aussehen:

```
[bootloader]  
timeout=8  
[operating systems]  
C:\grldr.mbr="Linux über GRUB4DOS"
```

4.3.4.3 Anmerkung zu openSUSE 11.0

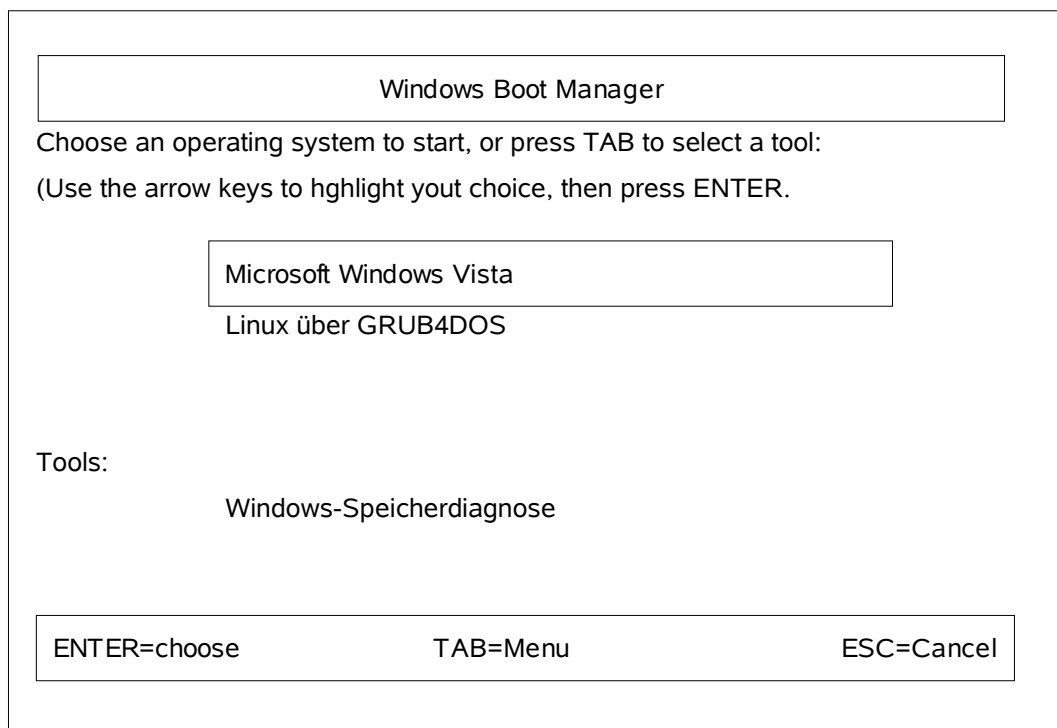
Die beschriebenen Vorgehensweisen funktionierten einwandfrei **vor** openSUSE 11.0.

Bei openSUSE 11.0 erkennt GRUB4DOS die in der Datei `menu.lst` aufgeführte Partition für das Verzeichnis `/boot` nicht.

Abhilfe:

- a) `vmlinuz` und `initrd` in das `root`-Verzeichnis einer anderen Partition kopieren und die `menu.lst` entsprechend anpassen. Es ist sogar möglich, dafür die NTFS-Partition `C:\` zu nehmen.
- b) openSUSE 11.0 über den `chainloader` starten, ggf. nach Tausch von `hd0` und `hd1` per `map`.

Vista zeigt dann nach dem Boot folgendes Menü an:



4.3.4.4 Optionale Schritte

`grldr` sucht automatisch nach einer `menu.lst` und zeigt sie an. Um das Suchen zu verkürzen kann man die `menu.lst` auch nach `C:\` kopieren.

Die folgenden Schritte unter Linux durchführen.

- Die Datei `/boot/grub/menu.lst` auf eine Partition kopieren, die von Windows gelesen werden kann.
- Diese Datei so ändern, dass sie in etwa folgenden Inhalt hat:

```
# Modified by ....., Windows/C, datum
default 0
timeout 8
```

```

title openSUSE extern
    root (hd1,4)
    kernel /boot/vmlinuz root=/dev/sda3 vga=0x31a \
    resume=/dev/sda2 splash=silent showopts
    initrd /boot/initrd

```

Für *hd1,4* ist die GRUB-Gerätebezeichnung für die Linux-Partition einzutragen, für *sda3* die Linux-Bezeichnung für die Linux-Partition und für *sda2* die Linux-Bezeichnung für die swap-Partition.

- Die so modifizierte **menu.lst** nach C:\ kopieren.

4.3.5 Wenn BIOS die externe Platte nicht erkennt

Es kann sein, dass das BIOS die externe Platte beim Booten nicht erkannt hat. Dann läuft GRUB ins Leere. In diesem Fall muss auf der internen Platte eine kleine Linux-Partition von etwa 30 MB angelegt werden, in die das Verzeichnis */boot* von der externen Platte kopiert wird. Die Aufrufe in der Datei *menu.lst* für Windows sind dann entsprechend anzupassen.

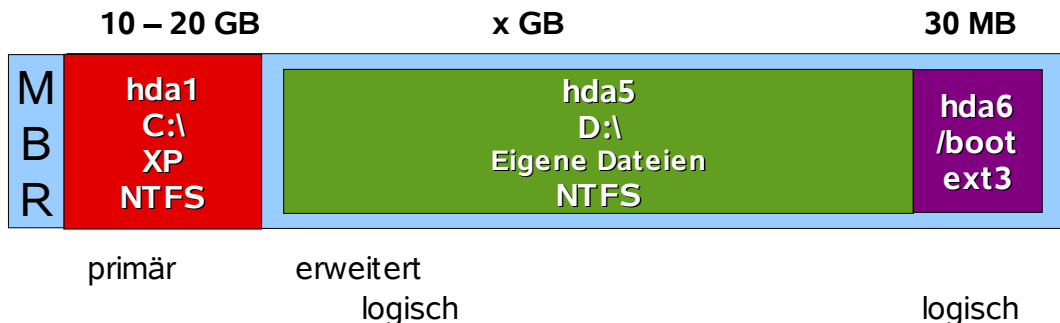
4.3.6 GRUB im MBR der internen Platte bei eigener Bootpartition für Linux auf der internen Platte

Wenn man für */boot* eine eigene Partition auf der internen Platte anlegt, kann man auch GRUB (bzw. dessen stage1) in den MBR der internen Platte schreiben.

Die eigene boot – Partition muss den Mountpoint */boot* erhalten.

Dann findet stage2 *menu.lst*, auch wenn die externe Platte nicht eingeschaltet ist.

Interne Platte:



Bevor man GRUB in den MBR installiert, kann man ihn sichern.

Der MBR der ersten bootbaren Festplatte liegt auf */dev/hda* oder */dev/sda*. Er lässt sich mit der Konsoleingabe sichern:

```
dd if=/dev/hda of=sicherungsdatei bs=512 count=1
```

Zum Zurückschreiben gibt man ein:

```
dd of=/dev/hda if=sicherungsdatei bs=512 count=1
```

Dabei wird allerdings auch die aktuelle Partitionstabelle mit dem Stand der Sicherung überschrieben. Will man das vermeiden, darf man nur 446 Byte zurückschreiben:

```
dd of=/dev/hda if=sicherungsdatei bs=446
```

Anmerkung zur Syntax:

dd eine Datei Konvertieren und kopieren
(convert and copy a file, hätte eigentlich mit *cc* abgekürzt werden sollen, *cc* war aber schon für den C-Compiler vergeben)

if= diese Datei lesen
of= in diese Datei schreiben
bs= x Byte konvertieren und schreiben
count= diese Anzahl von Eingabeblocks kopieren

Auch über

YaST / System / Konfiguration des Bootloaders
kann man den MBR wieder zurückschreiben.

