

# 3D Beschleunigung für ATI und NVIDIA einrichten



Bruno „Fuddl“ Kleinert <fuddl@tauware.de>



# Hilfreiches Grundwissen



- Bedienung des Paketmanagers
- Einfache Grundlagen der Kommandozeile



# Häufiges Einsteigerproblem



Symptom: Nach der GNU/Linux Installation funktioniert zwar die grafische Oberfläche aber 3D Anwendungen oder Spiele laufen nicht



# Der Haken bei ATI und NVIDIA



- normalerweise werden für ATI/NVIDIA Treiber ohne 3D Beschleunigung installiert
- ... weil es zwei Arten von Treibern gibt
  - quelloffene Treiber (ohne 3D)
  - proprietäre Treiber (mit 3D)
- quelloffene Treiber werden bei der Installation bevorzugt



# Der Haken bei ATI und NVIDIA



- ATI/NVIDIA rücken keine Spezifikationen zur Hardware heraus
  - **sehr** ärgerlich :-)
  - Entwicklung beschleunigter Treiber durch Community quasi unmöglich
  - ATI/NVIDIA bieten proprietäre Treiber an
- proprietäre Treiber selten im Hauptpaketarchiv
- meist in separatem Archiv für nicht-freie Software
  - z.B. bei Debian und Ubuntu der Fall



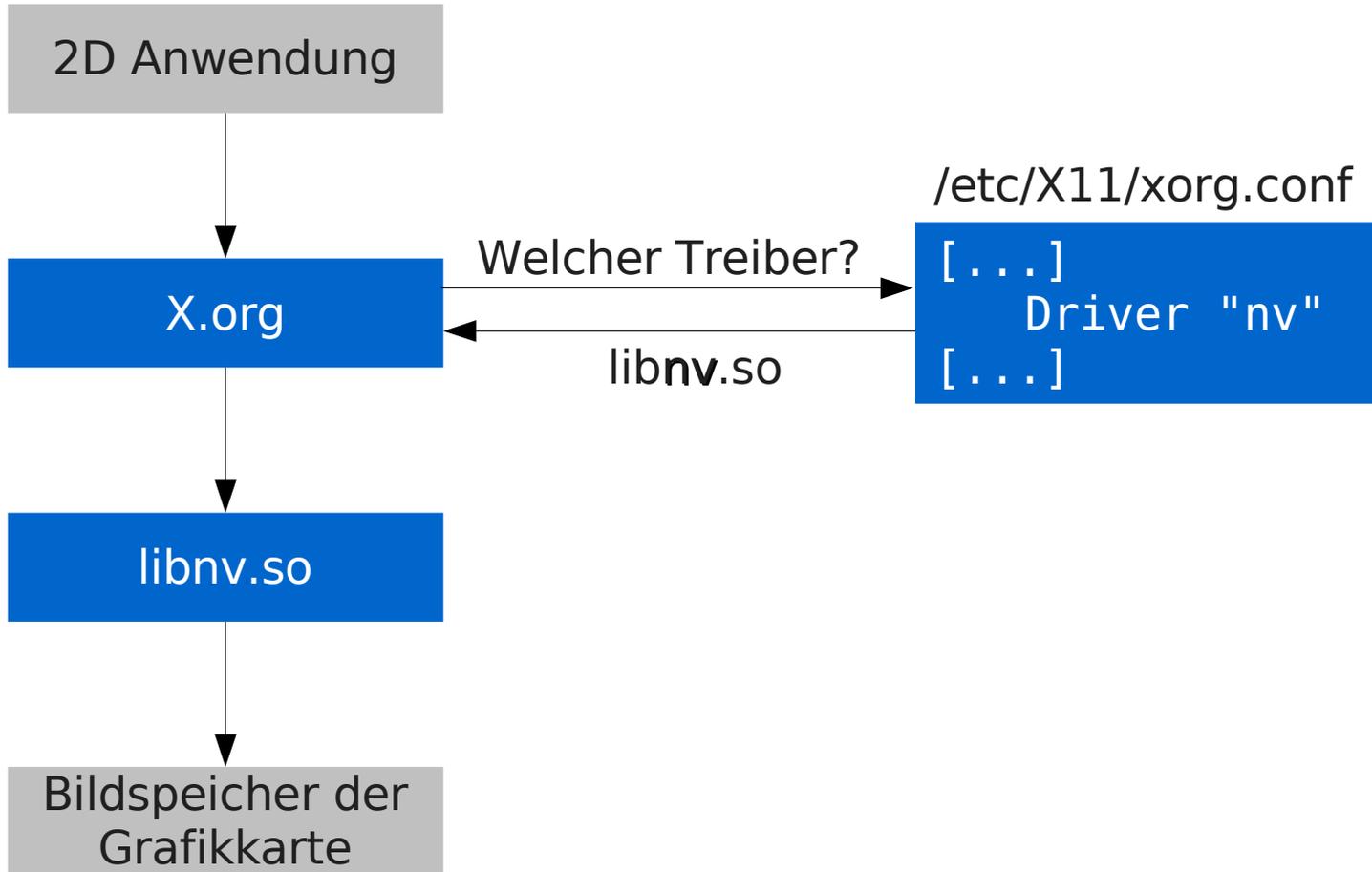
# Aufbau eines beschleunigten X.org Treibers



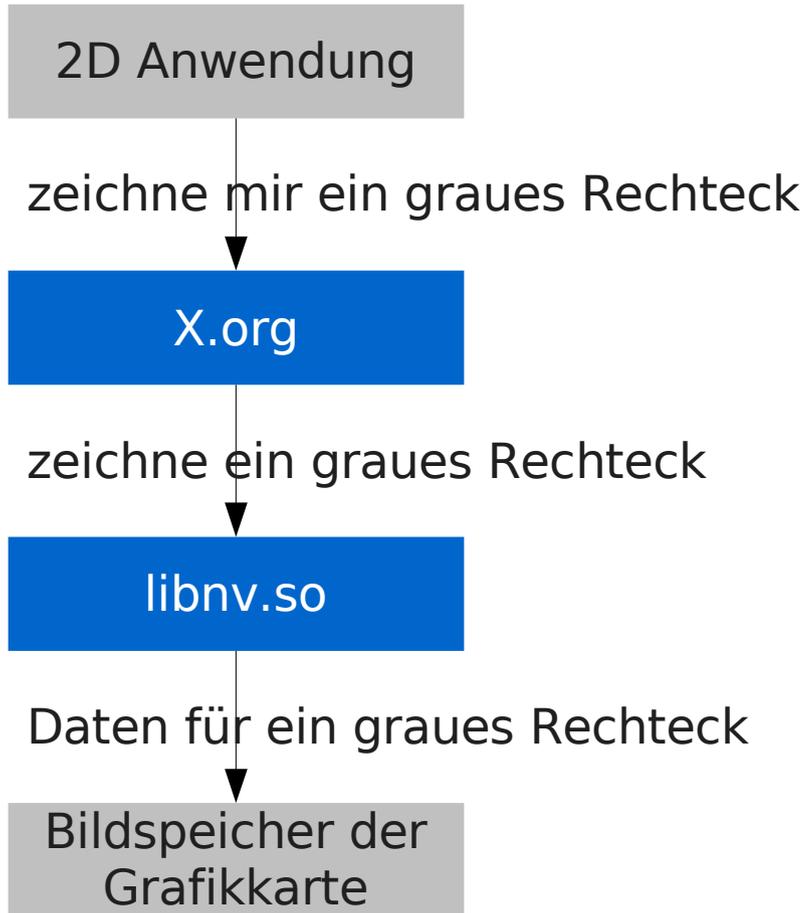
Wie funktioniert 2D und 3D Darstellung unter X.org?



# 2D ist noch einfach...



# So funktioniert...



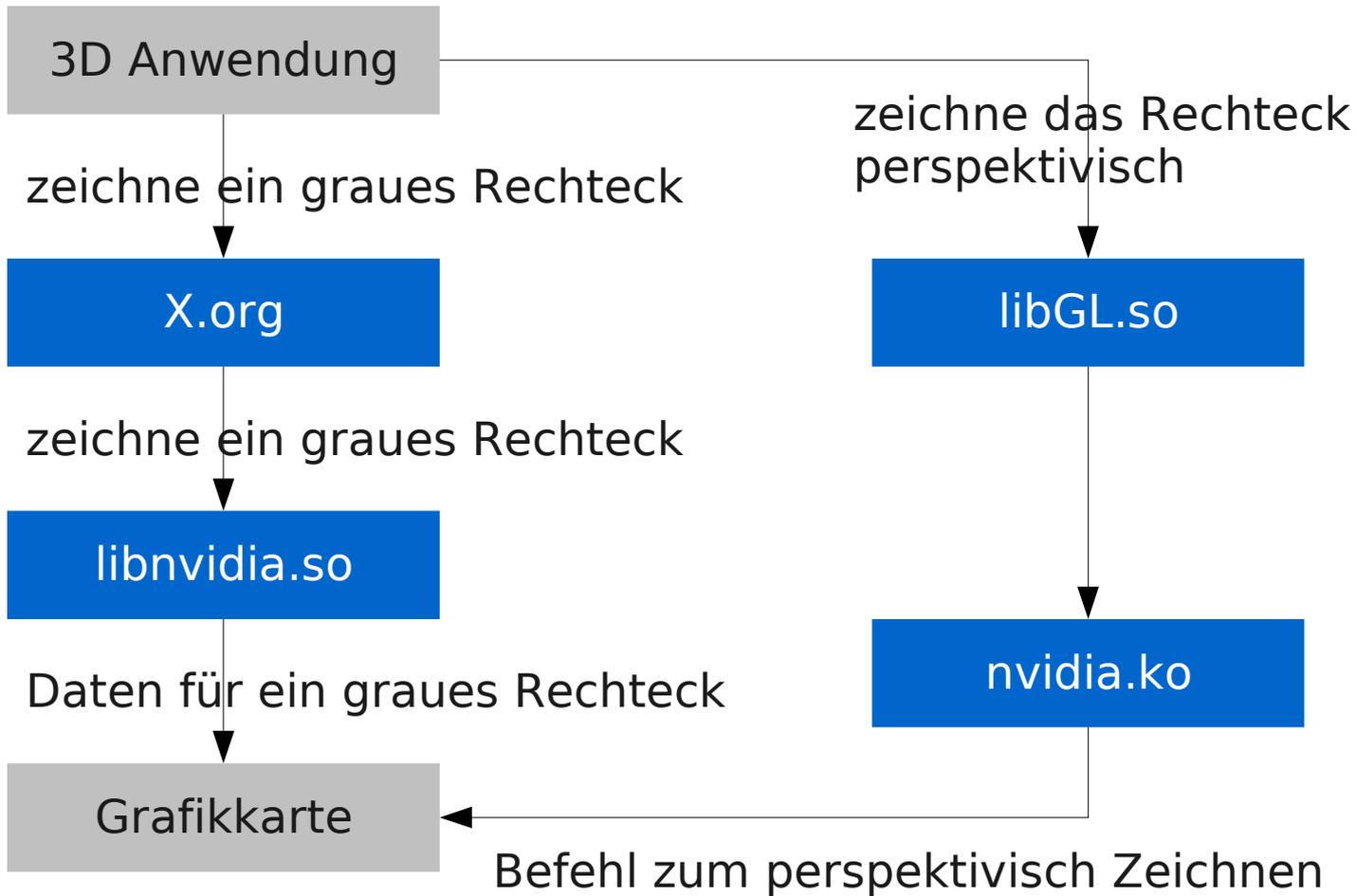
# 3D wird schwieriger...



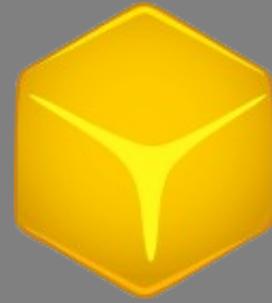
- unser graues Rechteck wollen wir perspektivisch anzeigen lassen!
- „Grafikkarte, zeichne das Rechteck perspektivisch“
  - der Grafikkarte müssen Kommandos geschickt werden können



# 3D wird schwieriger...



# Aufbau eines beschleunigten Grafikkartentreibers



- X.org Grafikkarten Bibliothek
  - steht nur vorkompiliert zur Verfügung
  - einfach als Paket installierbar
  - Paket hängt meistens ab von einem...
- ... Kernel Treibermodul
  - Verwaltet Zugriffe der Xserver auf die Karte
  - größtenteils als Quelltext verfügbar
  - Umstand: muss passend zum benutzten Linux Kern übersetzt werden



# Treiber mit 3D Beschleunigung

- bei ATI „fglrx“
- bei NVIDIA „nvidia“
  - für ältere NVIDIA Karten (TNT bis GeForce 2) „nvidia legacy“
- naheliegend: danach muss mit dem Paketmanager gesucht werden



# Im Hinterkopf behalten...

- beschleunigte Grafikkartentreiber bestehen aus zwei Teilen:
  1. Userspace Treiber (X.org Treiber/Bibliothek)
  2. Kernelspace Treiber (Kernel Modul)
    - nvidia(.ko)
    - fglrx(.ko)



# Welche Karte steckt im Rechner?

- PCI, AGP und PCI Express Karten erscheinen im OS als PCI Geräte
- lspci zeigt eingesteckte PCI Geräte an
- lspci | grep VGA zeigt im Rechner sitzende Grafikkarten an



# Wie man den Treiber NICHT installiert

- ATI und NVIDIA bieten hässliche Installationskripten an
  - umgehen Paketmanager
  - meistens latest-shoot-in-the-foot-technology :-)
- nur für **sehr** erfahrene Anwender
- höchstens für einen kurzen Test geeignet
- So nicht! NEIN! NEIN! NEIN!



# Wie man es sauber und richtig macht

- Treiber sind **nicht** abhängig vom Kartenmodell
  - Es kommt nur auf die Unterscheidung ATI, NVIDIA, NVIDIA legacy an
- mit dem Paketmanager muss nur nach `nvidia` oder `fglrx` gesucht werden
- Installation dieser Treiberpakete via Paketmanager



# Installation des Kerneltreibers



- für vorkompilierte Kernel der Distribution gibt es i.d.R. bereits vorkompilierte Treiber
  - einfach installierbar
- Gibt es keinen vorkompilierten Kerneltreiber, hilft oft ein Werkzeug der Distribution beim Kompilieren und der sauberen Installation des Treibers
  - z.B. bei Debian/Ubuntu der „module-assistant“



# Nach Installation der Treiberpakete



- X.org muss mitgeteilt werden den eben installierten Treiber zu verwenden
  - Konfiguration von X.org muss angepasst werden
    - `/etc/X11/xorg.conf`, Section "Device"
      - Driver "nv" durch Driver "nvidia" ersetzen
      - Driver "ati" durch Driver "fglrx" ersetzen



# Eine angepasste /etc/X11/xorg.conf



```
Section "Device"  
    BusID "1:0:0"  
    Driver "nvidia"  
    Identifier "gefurz_0"  
    Screen 0  
    Option "NoLogo" "1"  
EndSection
```



# Weitere Informationen



- Dokumentationen der Treiber geben Auskunft über weitere Einstellungen
- Dokumentation zu Treibern **unbedingt** lesen!!!



# Nach Anpassung der Konfiguration



- es ist **kein** Neustart des Rechners notwendig ;-)
- laufende Xsessions beenden
- im Falle von ATI
  - `modprobe -r radeon` (als root) deaktiviert eventuell geladene quelloffene Treiber
- X.org kann wieder gestartet werden
  - eventuell einfach `xdm/wdm/gdm/kdm` neu starten (als root)
- `glxinfo | grep direct` sollte jetzt Yes ausgeben



# Nützliche Werkzeuge für DRI/OpenGL



## → glxinfo

- Zeigt vom Treiber bereitgestellte OpenGL Versionen und Erweiterungen (GL\_EXT\_\*) an
- `glxinfo | grep direct` – zeigt an, ob DRI/OpenGL zur Verfügung steht



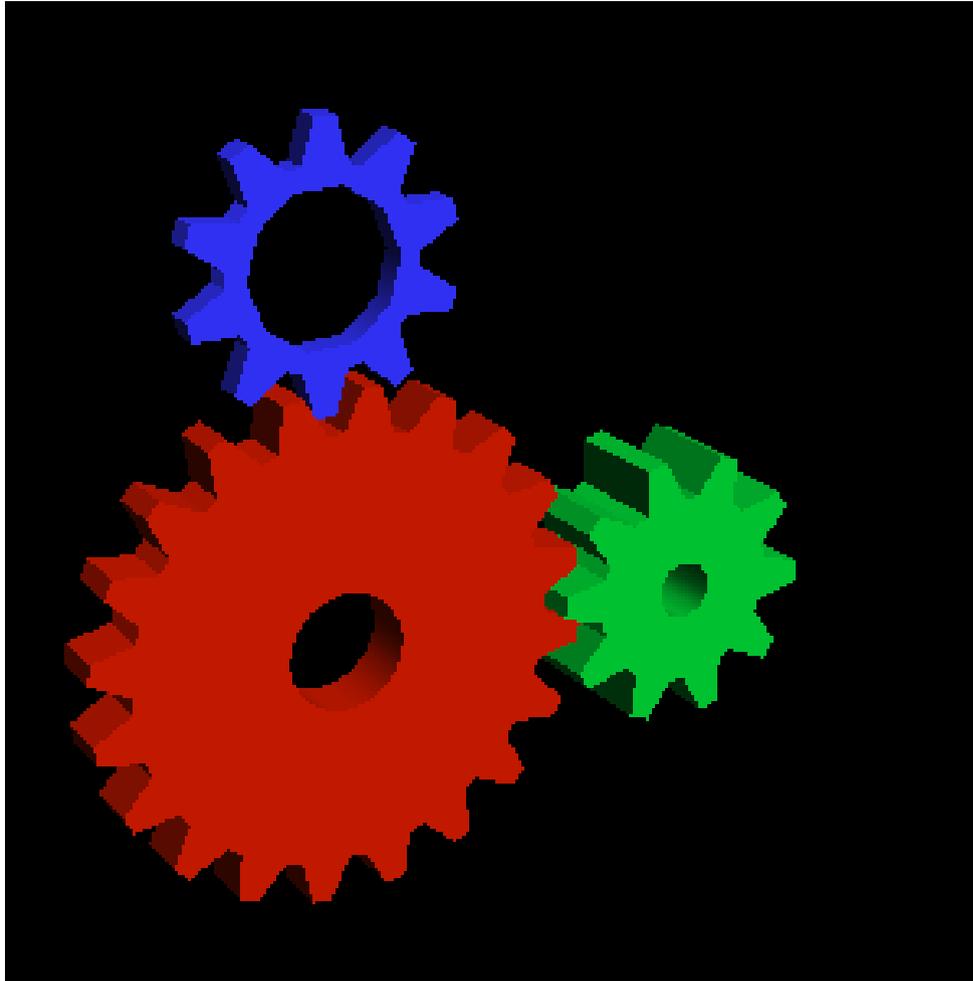
# glxinfo



```
$ glxinfo
name of display: :0.0
display: :0 screen: 0
direct rendering: Yes
server glx vendor string: NVIDIA Corporation
server glx version string: 1.4
server glx extensions:
    GLX_EXT_visual_info, GLX_EXT_visual_rating, GLX_SGIX_fbconfig,
    GLX_SGIX_pbuffer, GLX_SGI_video_sync, GLX_SGI_swap_control,
    GLX_EXT_texture_from_pixmap, GLX_ARB_multisample, GLX_NV_float_buffer
client glx vendor string: NVIDIA Corporation
client glx version string: 1.4
client glx extensions:
    GLX_ARB_get_proc_address, GLX_ARB_multisample, GLX_EXT_visual_info,
    GLX_EXT_visual_rating, GLX_EXT_import_context, GLX_SGI_video_sync,
    GLX_NV_swap_group, GLX_NV_video_out, GLX_SGIX_fbconfig, GLX_SGIX_pbuffer,
    GLX_SGI_swap_control, GLX_NV_float_buffer, GLX_ARB_fbconfig_float,
    GLX_EXT_fbconfig_packed_float, GLX_EXT_texture_from_pixmap,
    GLX_EXT_framebuffer_sRGB
GLX version: 1.3
GLX extensions:
    GLX_EXT_visual_info, GLX_EXT_visual_rating, GLX_SGIX_fbconfig,
    GLX_SGIX_pbuffer, GLX_SGI_video_sync, GLX_SGI_swap_control,
    GLX_EXT_texture_from_pixmap, GLX_ARB_multisample, GLX_NV_float_buffer,
```



# glxgears – DRI/OpenGL testen



# NVIDIA auf einem Debian etch



Installation des beschleunigten Treibers für eine  
NVIDIA Karte unter Debian bzw. Ubuntu



# NVIDIA Installation (mit Distributionskern)



1. Archiv `non-free` muss in `/etc/apt/sources.list` vorhanden sein (ggf. anschließend `aptitude update` ausführen)
2. `aptitude install nvidia-kernel-`uname -r`` (als root) installiert das nvidia Kernel Modul
3. `aptitude install nvidia-glx` (als root) installiert den X.org Treiber
4. Driver `"nv"` durch Driver `"nvidia"` ersetzen in `/etc/X11/xorg.conf`
5. laufende Xserver neu starten
6. fertig!



# NVIDIA Installation (ohne Distributionskern)



- Der Kerneltreiber muss kompiliert werden
  - falls man einen selbstkompilierten Kern einsetzt
  - falls es noch keine vorkompilierten Treiberpakete gibt (z.B. beim fglrx Kerneltreiber der Fall)



# NVIDIA Kerneltreiber mit dem „module-assistant“ installieren



- „module-assistant“ in Debian/Ubuntu
- Das Paket module-assistant installieren
  - `aptitude install module-assistant`
- Den Befehl `module-assistant` (als root) starten



# „module-assistant“



```
module-assistant, interaktiver Modus
Willkommen bei der Dialog-Oberfläche von module-assistant.
Diese
Benutzeroberfläche bietet den Zugang zu einigen Kommandos
des
Programms.

Um mehr zu erfahren, Wählen Sie die Option OVERVIEW.

Vor dem Beginn sollte die Funktion UPDATE einmal
ausgeführt

OVERVIEW Alle bekannten Kommandozeilen-Kommandos anzeigen
UPDATE   Interne Informationen über Pakete aktualisieren
PREPARE  Das System zum Kompilieren einrichten
SELECT   Zu bearbeitende Module/Quellcode auswählen
EXIT     Das Programm verlassen

<Ok>                <Abbrechen>
```



# Zur Auswahl verfügbarer Kerneltreiber



`module-assistant, interaktiver Modus`

Willkommen bei der Dialog-Oberfläche von module-assistant.  
Diese  
Benutzeroberfläche bietet den Zugang zu einigen Kommandos  
des  
Programms.

Um mehr zu erfahren, Wählen Sie die Option OVERVIEW.

Vor dem Beginn sollte die Funktion UPDATE einmal  
ausgeführt

OVERVIEW	Alle bekannten Kommandozeilen-Kommandos anzeigen
UPDATE	Interne Informationen über Pakete aktualisieren
PREPARE	Das System zum Kompilieren einrichten
<b>SELECT</b>	<b>Zu bearbeitende Module/Quellcode auswählen</b>
EXIT	Das Programm verlassen

<Ok>

<Abbrechen>



# Auswahl aus der Liste verfügbarer Kerneltreiber



```
module-assistant, Paketauswahl

Wählen Sie die interessanten Module (Quellcode) aus.
Pfeiltasten zum Navigieren, Leerzeile wählt aus, Return zum
weitermachen.
Abbrechen um ins Hautmenü zurückzukehren.

[ ] madwifi           Multiband Atheros Driver for WiFi
[ ] mga-vid           Kernel driver for the back-end scaler on Mat
[ ] misdn-kernel
[ ] ndiswrapper       ndiswrapper linux kernel module
[ ] nozomi             GlobeTrotter HSDPA kernel driver
[*] nvidia-kernel     NVIDIA binary kernel module
[ ] nvidia-kernel-legacy NVIDIA binary kernel module source (legacy v
[ ] openafs-modules   AFS distributed filesystem kernel module
[ ] openswan-modules  IPSEC kernel modules source for Openswan

<Ok>                <Abbrechen>
```



# Gewählten Kerneltreiber kompilieren



```
module-assistant, interaktiver Modus
Sie haben folgende Pakete ausgewählt:

nvidia-kernel

Wählen Sie die Kommandos, die durchgeführt werden sollen,

LIST   Installierte (binäre) Pakete auflisten
SEARCH Auflisten mit Suche mittels apt-cache
GET    Quellcode-Pakete holen oder aktualisieren
BUILD  Modul-Pakete kompilieren für den derzeitigen Kernel
INSTALL Pakete für den derzeitigen Kernel installieren
BACK   Zurück zur Paket-Auswahl

      <Ok>                <Abbrechen>
```



# „module-assistant“ holt fehlende Pakete selbst



```
module-assistant, Paketinstallation  
Das Quellcode-Paket wurde wohl nicht installiert.  
Möchten Sie dieausgewählten Pakete jetzt  
installieren/aktualisieren?  
  
<Ja> <Nein>
```



# Kompiliervorgang



# Installation des kompilierten Kerneltreibers



```
module-assistant, Paketinstallation
Sollen die gebauten Modul-Pakete jetzt installiert
<Ja> <Nein>
```



# „module-assistant“ beenden



```
module-assistant, interaktiver Modus
Sie haben folgende Pakete ausgewählt:

nvidia-kernel

Wählen Sie die Kommandos, die durchgeführt werden sollen,

LIST  Installierte (binäre) Pakete auflisten
SEARCH  Auflisten mit Suche mittels apt-cache
GET     Quellcode-Pakete holen oder aktualisieren
BUILD   Modul-Pakete kompilieren für den derzeitigen Kernel
INSTALL Pakete für den derzeitigen Kernel installieren
BACK    Zurück zur Paket-Auswahl

      <Ok>                <Abbrechen>
```



# „module-assistant“ beenden



```
module-assistant, interaktiver Modus
Willkommen bei der Dialog-Oberfläche von module-assistant.
Diese
Benutzeroberfläche bietet den Zugang zu einigen Kommandos
des
Programms.

Um mehr zu erfahren, Wählen Sie die Option OVERVIEW.

Vor dem Beginn sollte die Funktion UPDATE einmal
ausgeführt

OVERVIEW Alle bekannten Kommandozeilen-Kommandos anzeigen
UPDATE   Interne Informationen über Pakete aktualisieren
PREPARE  Das System zum Kompilieren einrichten
SELECT   Zu bearbeitende Module/Quellcode auswählen
EXIT     Das Programm verlassen

<Ok>           <Abbrechen>
```



# Unterschiede bei einer ATI Installation



- Es gibt keine vorkompilierten Pakete des Kerneltreibers
  - es muss der „module-assistant“ verwendet werden
- Das Paket des X.org Treibers heißt „fglrx-driver“
  - `aptitude install fglrx-driver`
- `/etc/X11/xorg.conf`
  - Section "Device"
    - Driver "fglrx" anstatt Driver "ati"/"radeon"



# ATI mit dem „module-assistant“



```
module-assistant, Paketauswahl

Wählen Sie die interessanten Module (Quellcode) aus.
Pfeiltasten zum Navigieren, Leerzeile wählt aus, Return zum
weitermachen.
Abbrechen um ins Hautmenü zurückzukehren.

[ ] dvb-driver
[ ] e100
[ ] eagle-usb-modules   EAGLE-USB binary kernel module source for Ea
[ ] em8300             Kernel module for DXR3/Hollywood+ decoder ca
[ ] exmap-modules     determine how much physical memory and swap
[*] fglrx-kernel       kernel module source for the ATI graphics ac
[ ] freeswan-modules  IPSEC kernel modules transition package to 0
[ ] ftape
[ ] ftpfs

<Ok>                <Abbrechen>
```



# Faustregel zur Installation



1. Installation des Kerneltreibers
2. Installation des X.org Treibers
3. Anpassen der X.org Konfiguration
4. Neustart aller laufenden Xserver
5. fertig!



# Hilfreiche Programme (NVIDIA)



- nvidia-settings (X Anwendung)
- nvclock (Textbasiert, **NUR für erfahrene Benutzer!**)



# nvidia-settings



The screenshot shows the nvidia-settings application window. On the left is a sidebar with a tree view containing the following items: 'tittchen:0.0' (expanded), 'X Server Color Correction', 'X Server XVideo Settings', 'Cursor Shadow', 'OpenGL Settings', 'OpenGL/GLX Information', 'Antialiasing Settings' (highlighted), 'Display Devices' (expanded), 'MED MD6155AN', 'AMW M179D', and 'nvidia-settings Configuration'. The main area features a header with a green background, a magnifying glass over a globe, and the NVIDIA logo. Below the header are three settings panels: 'Antialiasing Settings' with an unchecked 'Override Application Setting' checkbox and a slider set to 'Off'; 'Anisotropic Filtering' with an unchecked 'Override Application Setting' checkbox and a slider set to '1x'; and 'Texture Quality' with an unchecked 'Texture Sharpening' checkbox. At the bottom right are two buttons: 'Hilfe' (Help) and 'Beenden' (Quit).



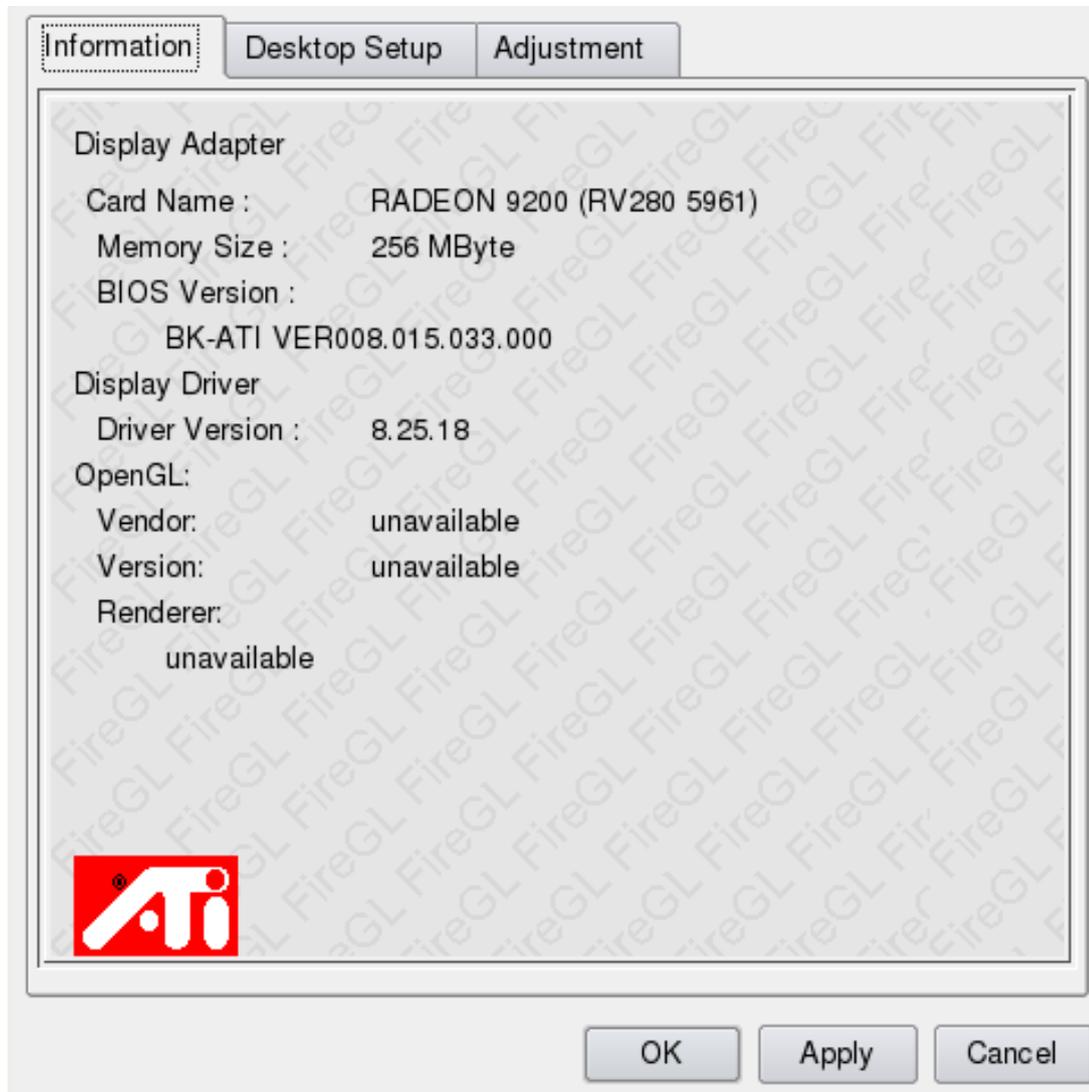
# Hilfreiche Programme (ATI)



- fireglcontrol (X Anwendung)
- rovclock (Textbasiert, **NUR für erfahrene Benutzer!**)



# fireglcontrol



# Kritik an ATI/NVIDIA Treiber

- fglrx
  - qualitativ minderwertig
  - teilweise lange Wartezeit auf Unterstützung neuer Karten
- nvidia
  - neue Karten werden sehr schnell unterstützt
- nvidia als auch fglrx sind größtenteils closed source Software
  - nur „Mainstream“ Fehler werden schnell behoben
  - schlechte/keine suspend Unterstützung
  - Wartezeiten auf Anpassungen an neue Kernel Versionen



# 3D Beschleunigung für ATI und NVIDIA einrichten



Fragen?



# „Known Bugs“



- Anpassen der xorg.conf bei NVIDIA
  - Entfernen von Load „dri“, „GLcore“ fehlt (Danke Andreas Berger)
- Zusätzliche Hilfsprogramme
  - Externe Programme für TV Ausgänge erwähnen (Danke Benjamin Klier)
- module-assistant
  - Ein paar Schritte vergessen

