# Fedora 11 nVidia Twinview Support

**Finnbarr P. Murphy**
*(fpm@fpmurphy.com)*

Fedora 11 (*Leonidas*) ships with the *nouveau* nVidia graphics driver preloaded by default if a nVidia graphics card is detected at install time. Previous versions of Fedora used the older X.Org *nv* driver.

The *nouveau* project aims at producing Open Source 3D drivers for nVidia graphics cards. According to the nouveau project Wiki

> 2D-support is in fairly good shape with EXA acceleration, Xv and Randr12 (think of dual-head, rotations, etc.). Randr12 should work for all cards up to, and including, Geforce 9000 series, although some issues with Geforce 8/9 laptops may still exist, for such issues bug reports should be submitted. Randr12 is now the default. Any 3D functionality that might exist is still unsupported, do not ask for instructions to try it. Also, VT switching while X is running is considered lucky."

Well, I certainly quickly ran into the VT switching issue! It worked but not consistently.

Unfortunately the *nouveau* driver currently does not support nVidia TwinView functionality and I suspect that it will be a long time before it does if ever!

To use TwinView with Fedora 11, you have to load the correct nVidia drivers from rpmfusion.org. I described how to do this in detail in a previous post so I will not repeat that information here.

You also need to modify your *grub.conf* file to include the *nopat* kernel boot option as shown below.

```
title Fedora (2.6.29.4-167.fc11.x86_64)
       root (hd0,1)
       kernel /vmlinuz-2.6.29.4-167.fc11.x86_64 ro root=/dev/mapper/vg_ultra-lv_root rhgb
quiet nopat
       initrd /initrd-2.6.29.4-167.fc11.x86_64.img
```

The *nopat* option is needed for this particular kernel (2.6.29.4) as it appears to still have broken *PAT* functionality.

For those readers who are unaware of what *PAT* is, here is a brief explanation. Traditionally page caching was controlled by a CPU feature called *Memory Type Range Registers (MTRR)*. A CPU has a finite and limited set of *MTRR*s each of which control part of the physical address space. To overcome this limitation and provide a more flexible architecture, Intel and other x86 CPU vendors added a set of bits to page table entries to control how a CPU does page caching. These bits are called the *Page Attribute Table (PAT)*. Incidentally, the 2.6.26 kernel was the first Linux kernel to support PATs.

Unless you rebuild your initial ramdisk (*initrd*), the *nouveau* driver will remain loaded in the kernel. I prefer not to have the *nouveau* driver loaded in my kernel if I am not using it so I added *nouveau* to the list of blacklisted drivers in */etc/modprobe.d/blacklist.conf* and rebuild *initrd*.

```
# mv /boot/initrd-`uname -r`.img /boot/initrd-`uname -r`.img.backup
# mkinitrd -v /boot/initrd-`uname -r`.img `uname -r`
Creating initramfs
```



  

```
Looking for driver for /dev/mapper/vg_ultra-lv_root in /sys/block/dm-0
Found DeviceMapper component dm-0
Looking for deps of module scsi:t-0x00
Looking for deps of module pci:v00008086d00002922sv00008086sd00005442bc01sc06i01
Looking for driver for /dev/mapper/vg_ultra-lv_swap in /sys/block/dm-1
Found DeviceMapper component dm-1
Using modules:
Building initrd in /tmp/initrd.txR0Kd
/sbin/nash -> /tmp/initrd.txR0Kd/bin/nash
  /usr/lib64/libnash.so.6.0.86 -> /tmp/initrd.txR0Kd/usr/lib64/libnash.so.6.0.86
  /usr/lib64/libbdevid.so.6.0.86 -> /tmp/initrd.txR0Kd/usr/lib64/libbdevid.so.6.0.86
  /lib64/libdevmapper.so.1.02 -> /tmp/initrd.txR0Kd/lib64/libdevmapper.so.1.02
  /lib64/libparted-1.8.so.8 -> /tmp/initrd.txR0Kd/lib64/libparted-1.8.so.8
    /lib64//libparted-1.8.so.8.0.0 -> /tmp/initrd.txR0Kd/lib64//libparted-1.8.so.8.0.0
  /lib64/libblkid.so.1 -> /tmp/initrd.txR0Kd/lib64/libblkid.so.1
    /lib64//libblkid.so.1.0 -> /tmp/initrd.txR0Kd/lib64//libblkid.so.1.0
  /lib64/libselinux.so.1 -> /tmp/initrd.txR0Kd/lib64/libselinux.so.1
  /lib64/libsepol.so.1 -> /tmp/initrd.txR0Kd/lib64/libsepol.so.1
  /lib64/libuuid.so.1 -> /tmp/initrd.txR0Kd/lib64/libuuid.so.1
    /lib64//libuuid.so.1.2 -> /tmp/initrd.txR0Kd/lib64//libuuid.so.1.2
  /lib64/libpopt.so.0 -> /tmp/initrd.txR0Kd/lib64/libpopt.so.0
    /lib64//libpopt.so.0.0.0 -> /tmp/initrd.txR0Kd/lib64//libpopt.so.0.0.0
  /lib64/libresolv.so.2 -> /tmp/initrd.txR0Kd/lib64/libresolv.so.2
    /lib64//libresolv-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//libresolv-2.10.1.so
      /lib64/libc.so.6 -> /tmp/initrd.txR0Kd/lib64/libc.so.6
        /lib64//libc-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//libc-2.10.1.so
          /lib64/ld-linux-x86-64.so.2 -> /tmp/initrd.txR0Kd/lib64/ld-linux-x86-64.so.2
           /lib64//ld-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//ld-2.10.1.so
  /lib64/libdl.so.2 -> /tmp/initrd.txR0Kd/lib64/libdl.so.2
    /lib64//libdl-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//libdl-2.10.1.so
  /usr/lib64/libelf.so.1 -> /tmp/initrd.txR0Kd/usr/lib64/libelf.so.1
    /usr/lib64//libelf-0.141.so -> /tmp/initrd.txR0Kd/usr/lib64//libelf-0.141.so
  /usr/lib64/libnl.so.1 -> /tmp/initrd.txR0Kd/usr/lib64/libnl.so.1
    /usr/lib64//libnl.so.1.1 -> /tmp/initrd.txR0Kd/usr/lib64//libnl.so.1.1
  /lib64/libm.so.6 -> /tmp/initrd.txR0Kd/lib64/libm.so.6
    /lib64//libm-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//libm-2.10.1.so
  /lib64/libgcc_s.so.1 -> /tmp/initrd.txR0Kd/lib64/libgcc_s.so.1
    /lib64//libgcc_s-4.4.0-20090506.so.1 -> /tmp/initrd.txR0Kd/lib64//libgcc_s-4.4.0-20090
506.so.1
  /lib64/libreadline.so.5 -> /tmp/initrd.txR0Kd/lib64/libreadline.so.5
    /lib64//libreadline.so.5.2 -> /tmp/initrd.txR0Kd/lib64//libreadline.so.5.2
  /lib64/librt.so.1 -> /tmp/initrd.txR0Kd/lib64/librt.so.1
    /lib64//librt-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//librt-2.10.1.so
      /lib64/libpthread.so.0 -> /tmp/initrd.txR0Kd/lib64/libpthread.so.0
        /lib64//libpthread-2.10.1.so -> /tmp/initrd.txR0Kd/lib64//libpthread-2.10.1.so
  /lib64/libtinfo.so.5 -> /tmp/initrd.txR0Kd/lib64/libtinfo.so.5
    /lib64//libtinfo.so.5.7 -> /tmp/initrd.txR0Kd/lib64//libtinfo.so.5.7
/sbin/modprobe -> /tmp/initrd.txR0Kd/bin/modprobe
/sbin/rmmod -> /tmp/initrd.txR0Kd/bin/rmmod
resolving for MODULES
and that has items of
resolving for availmodules
and that has items of
/sbin/lvm -> /tmp/initrd.txR0Kd/bin/lvm
/etc/lvm -> /tmp/initrd.txR0Kd/etc/lvm
`/etc/lvm/lvm.conf' -> `/tmp/initrd.txR0Kd/etc/lvm/lvm.conf'
/etc/sysconfig/keyboard -> /tmp/initrd.txR0Kd/etc/sysconfig/keyboard
/bin/loadkeys -> /tmp/initrd.txR0Kd/bin/loadkeys
/lib/kbd/keymaps/i386/qwerty/us.map.gz -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/i386/qwerty/u
s.map.gz
/lib/kbd/keymaps/i386/include/qwerty-layout.inc -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/i386
/include/qwerty-layout.inc
/lib/kbd/keymaps/i386/include/compose.inc -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/i386/inclu
de/compose.inc
/lib/kbd/keymaps/include/compose.latin4 -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/comp
ose.latin4
/lib/kbd/keymaps/include/compose.8859_8 -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/comp
ose.8859_8
```



  

```
/lib/kbd/keymaps/include/compose.latin1 -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/comp
ose.latin1
/lib/kbd/keymaps/include/compose.latin3 -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/comp
ose.latin3
/lib/kbd/keymaps/include/compose.8859_7 -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/comp
ose.8859_7
/lib/kbd/keymaps/include/compose.latin2 -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/comp
ose.latin2
/lib/kbd/keymaps/include/compose.latin -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/include/compo
se.latin
/lib/kbd/keymaps/i386/include/linux-with-alt-and-altgr.inc -> /tmp/initrd.txR0Kd/lib/kbd/k
eymaps/i386/include/linux-with-alt-and-altgr.inc
/lib/kbd/keymaps/i386/include/linux-keys-bare.inc -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/i3
86/include/linux-keys-bare.inc
/lib/kbd/keymaps/i386/include/euro1.map.gz -> /tmp/initrd.txR0Kd/lib/kbd/keymaps/i386/incl
ude/euro1.map.gz
/etc/sysconfig/i18n -> /tmp/initrd.txR0Kd/etc/sysconfig/i18n
/bin/setfont -> /tmp/initrd.txR0Kd/bin/setfont
/lib/kbd/consolefonts/latarcyrheb-sun16.psfu.gz -> /tmp/initrd.txR0Kd/lib/kbd/consolefonts
/latarcyrheb-sun16.psfu.gz
/lib/udev/console_init -> /tmp/initrd.txR0Kd/lib/udev/console_init
  /lib64/libglib-2.0.so.0 -> /tmp/initrd.txR0Kd/lib64/libglib-2.0.so.0
    /lib64//libglib-2.0.so.0.2000.1 -> /tmp/initrd.txR0Kd/lib64//libglib-2.0.so.0.2000.1
probing for modules for drm device card0
Adding graphics device card0
Looking for deps of module pci:v000010DEd00000640sv00003842sd0000C959bc03sc00i00: i2c-core
 nvidia
Adding module i2c-core
Adding module nvidia
resolving for GRAPHICSMODS
and that has items of i2c-core nvidia
Looking for deps of module i2c-core
Looking for deps of module nvidia: i2c-core
copy from `/lib/modules/2.6.29.4-167.fc11.x86_64/kernel/drivers/i2c/i2c-core.ko' [elf64-x8
6-64] to `/tmp/initrd.txR0Kd/lib/modules/2.6.29.4-167.fc11.x86_64/i2c-core.ko' [elf64-x86-
64]
copy from `/lib/modules/2.6.29.4-167.fc11.x86_64/extra/nvidia-173xx/nvidia.ko' [elf64-x86-
64] to `/tmp/initrd.txR0Kd/lib/modules/2.6.29.4-167.fc11.x86_64/nvidia.ko' [elf64-x86-64]
/sbin/plymouthd -> /tmp/initrd.txR0Kd/bin/plymouthd
........
........
Adding module scsi_wait_scan
copy from `/lib/modules/2.6.29.4-167.fc11.x86_64/kernel/drivers/scsi/scsi_wait_scan.ko' [e
lf64-x86-64] to `/tmp/initrd.txR0Kd/lib/modules/2.6.29.4-167.fc11.x86_64/scsi_wait_scan.ko
' [elf64-x86-64]
This initrd uses dynamic shared objects.
Adding dynamic linker configuration files.
/etc/ld.so.conf -> /tmp/initrd.txR0Kd/etc/ld.so.conf
/etc/ld.so.conf.d/kernel-2.6.29.4-167.fc11.x86_64.conf -> /tmp/initrd.txR0Kd/etc/ld.so.con
f.d/kernel-2.6.29.4-167.fc11.x86_64.conf
/etc/ld.so.conf.d/mysql-x86_64.conf -> /tmp/initrd.txR0Kd/etc/ld.so.conf.d/mysql-x86_64.co
nf
/etc/ld.so.conf.d/nvidia-lib64.conf -> /tmp/initrd.txR0Kd/etc/ld.so.conf.d/nvidia-lib64.co
nf
/etc/ld.so.conf.d/xulrunner-64.conf -> /tmp/initrd.txR0Kd/etc/ld.so.conf.d/xulrunner-64.co
nf
/etc/ld.so.conf.d/qt-x86_64.conf -> /tmp/initrd.txR0Kd/etc/ld.so.conf.d/qt-x86_64.conf
Running ldconfig
#
```

After rebooting your system, if you use *dmesg* or *lsmod*, you will see that the *nvidia* driver was loaded instead of the *nouveau* driver.

You will also see that for some reason Plymouth no longer runs with a graphical splash screen if it previously did so. Plymouth is the replacement for the old RedHat Graphical Boot (RHGB). It was written by Ray Strode, Kristian Hogsberg and Peter Jones of Redhat and first shipped in

  

`Fedora 10.`

Finally, you do not need to modify your *`xorg.conf`* file for `Fedora 11`. It should just work.



  