Nouveau

Recap, on-going and future work

Martin Peres, Lucas Stach & the Nouveau community

Ph.D. student at LaBRI, B.Eng. student at HfTL

February 3, 2012

Summary

Your host for the next hour

- Martin Peres (mupuf)
- Maarten Lankhorst (mlankhorst)
- Lucas Stach (lynxeye)

Also attending

- Emil Velikov (xexaxo)
- Francisco Jerez (curro)
- Roy Spliet (rspliet)

Summary

History: NVIDIA – a new hope

- 1998(?): NVIDIA releases a Linux open-source 2D-only driver(nv)
- 1998: Obfuscation commit (release only pre-processed source)

After we already finalized XFree86-3.3.3 NVIDIA forced The XFree86 Project to replace the sources we had with sources that were partly run through the C preprocessor in order to remove some of the names that NVIDIA thought might give away IP from NVIDIA. This resulted in unreadable and unmaintainable code.

The XFree86 Project is strongly opposed to such obfuscated code. We do not regard this as free software according to our standards. Due to the extremely late date of this decision from NVIDIA we decided to include the code as offered by NVIDIA. We are considering to remove support for the later NVIDIA chips in a future release, though.

Xfree86, commit message from 11/18/98

History: The open-source strikes back

- 2005: Stephane Marchesin improves nv and works on 3D
  - Project named Nouveau after an unfortunate automatic spelling correction
  - Back-port nv updates to Nouveau
- 2008: Open Arena runs on nv40
- 2009: KMS driver based on TTM
- 2010: Merged in Linux 2.6.33
- 2010: Nv is deprecated by NVIDIA, say “use VESA”.

History: The return of the Jedi

"It's so hard to write a graphics driver that open-sourcing it would not help [...] In addition, customers aren't asking for opensource drivers."
Andrew Fear, NVIDIA software product manager, April 2006

Summary

Chipset families

- NV03: RIVA 128
- NV04: RIVA TNT/TNT2
- NV10: Geforce2/4 MX
- NV20: Geforce 3, Geforce 4 Ti
- NV30: Geforce FX
- NV40: Geforce 6/7 (the first real family)
- NV50: Geforce 8/9/100/200/300 (the biggest family)
- NVC0: Geforce 400/500, AKA Fermi
- NVD9: Strange Fermi

Hardware Introduction

- NVIDIA GPUs are controlled by "objects"
- command submission by DMA buffers
- GPU can accesss system memory through the aperture
- NV50 features a full blown virtual memory system

important engines

- PFIFO: main command fetching engine
- PCRTC/PDISPLAY: display scanout engines
- PGRAPH: main graphics object AKA OpenGL cast into silicon
- PMPEG: video acceleration found on older cards (not much use today)
- PBSP/PVP/PPPP: modern video decoding engines

The 4 parts of Nouveau

- Linux Kernelmodule
- libdrm-nouveau
- DDX: xf86-video-nouveau
- Mesa3D drivers:
  - nouveau-vieux
  - nvfx
  - nv50
  - nvc0

Linux kernel module Nouveau

- Resource management
  - GPU Channels
  - Memory Management
- Command-submission
- Kernel Mode Setting
- Power Management

Power management

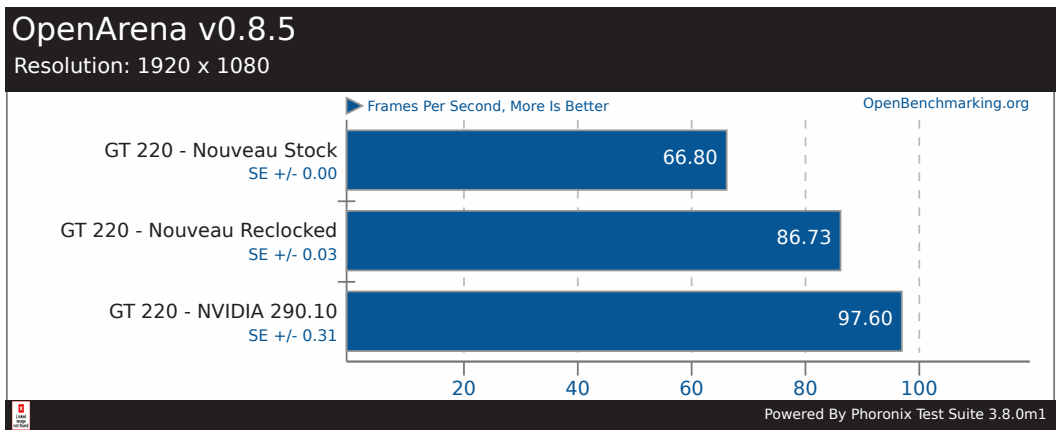
- Parsing power-management table from the vbios
- Temperature management: thermal calibration + fan management
- Setting the clocks: changes the performance level
- Other: clock gating, PCIE downclocking, DAC reclocking

current state

- nvc0: setting engines clocks (mostly)
- nv40-d9: memory timings (almost ready)
- nv30-c0: some kind of support to reclock (unreliable)
- nv50-a3: reliable clock changes (almost)
- fan management: toogle (todo), pwm (done), i2c (WIP)
- performance counters (initial work on nv40-c0)

WIP

- nva3-d9: setting memory clocks
- nv50-a3: fixing bugs
- nv50-d9: getting stable reclocking
- AGP/PCIE, clock gating: reverse engineering
- Fan: temperature-based management (toogle, pwm and i2c)
- perform infrastructure, dynamic reclocking



http://www.phoronix.com/scan.php?page=article&item=nouveau_reclocking_one&num=5

libdrm-nouveau

- buffer management
  - everything is a buffer!
- wraps around the IOCTL interface

work in progress

- currently rewritten
- designed with nv40 class hardware in mind
- pushbuffer replay

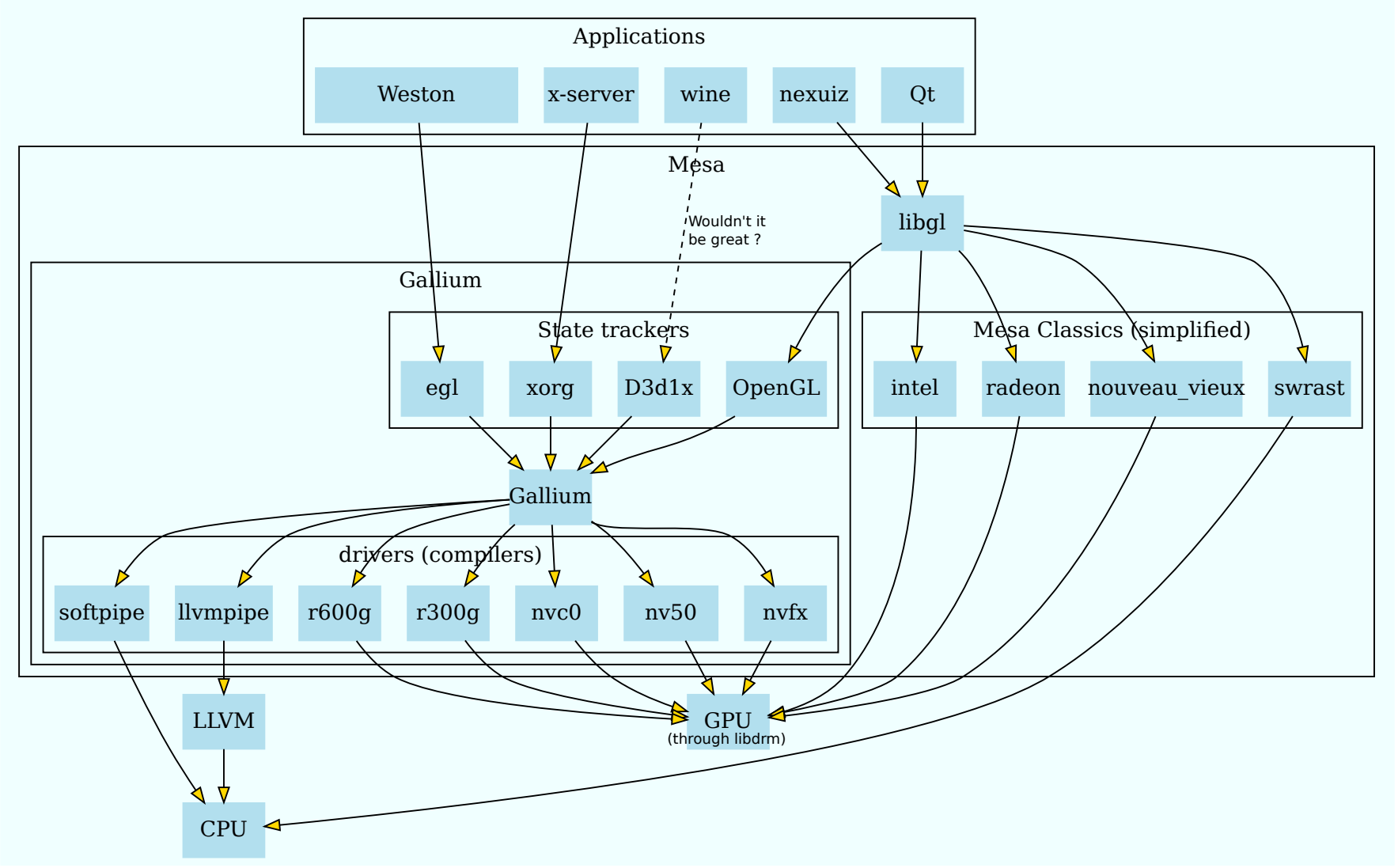
DDX – xf86-video-nouveau

- EXA (2D acceleration)
- X-Video
- supports full range of GPUs (NV04 - NVD9)
- makes use of all engines for acceleration (including 3D engine)

Mesa3D: drivers for 3D and more

- two different approaches here: Classic and Gallium3D
- nouveau-vieux: classic Mesa3D driver for NV04,NV1x,NV2x
  - only supports fixed function OpenGL
- Gallium pipe-driver
  - nvfx
  - nv50
  - nvc0

Components


Applications
Weston
x-server
wine
nexuiz
Qt
Mesa
Wouldn't it be great ?
libgl
Gallium
State trackers
egl
xorg
D3d1x
OpenGL
Mesa Classics (simplified)
intel
radeon
nouveau_vieux
swrast
Gallium
drivers (compilers)
softpipe
llvmpipe
r600g
r300g
nvc0
nv50
nvfx
LLVM
GPU
(through libdrm)
CPU

nvfx

- Gallium driver for NV3x,4x class GPUs
- one of the first gallium drivers
- created by merging separate nv30 and nv40 drivers
- accumulated much old cruft
- no maintainer for over one year

work in progress

- will hopefully be soon replaced by rewritten driver

nv50

- Gallium driver for NV50 class GPUs
- current codebase was formed by adapting nvc0
- maybe first free software driver to allow hardware OpenCL

work in progress

- merge reworked shader compiler
- implement remaining OpenGL 3 features
- merge OpenCL work

nvc0

- nvc0: Gallium driver for NVC0 class GPUs
- one of the first drivers to support OpenGL 3

work in progress

- working towards features in DX11 and OGL3+

```
    GLX_SGIX_fbconfig, GLX_SGIX_pbuffer, GLX_SGIX_visual_select_group,
    GLX_EXT_texture_from_pixmap
OpenGL vendor string: nouveau
OpenGL renderer string: Gallium 0.4 on NVC4
OpenGL version string: 3.0 Mesa 8.0-devel (git-8cbe699)
OpenGL shading language version string: 1.30
OpenGL extensions:
    GL_ARB_multisample, GL_EXT_abgr, GL_EXT_bgra, GL_EXT_blend_color,
    GL_EXT_blend_minmax, GL_EXT_blend_subtract, GL_EXT_copy_texture,
    GL_EXT_polygon_offset, GL_EXT_subtexture, GL_EXT_texture_object,
    GL_EXT_vertex_array, GL_EXT_compiled_vertex_array, GL_EXT_texture,
    GL_EXT_texture3D, GL_IBM_rasterpos_clip, GL_ARB_point_parameters,
    GL_EXT_draw_range_elements, GL_EXT_packed_pixels, GL_EXT_point_parameters,
```

OpenCL

- developed by Francisco Jerez
- sponsored by the X.org EVOC
- very experimental
- in-depth presentation here at 18:00

d3d1x

- Direct 3D 10/11 on Gallium
- "works" only on nouveau, still a bit hackish
- developed by nv50/c0's maintainer: Christoph Bumiller
- Unigine Heaven runs ont it when using a modified version of wine

Video decoding

- What's the point?

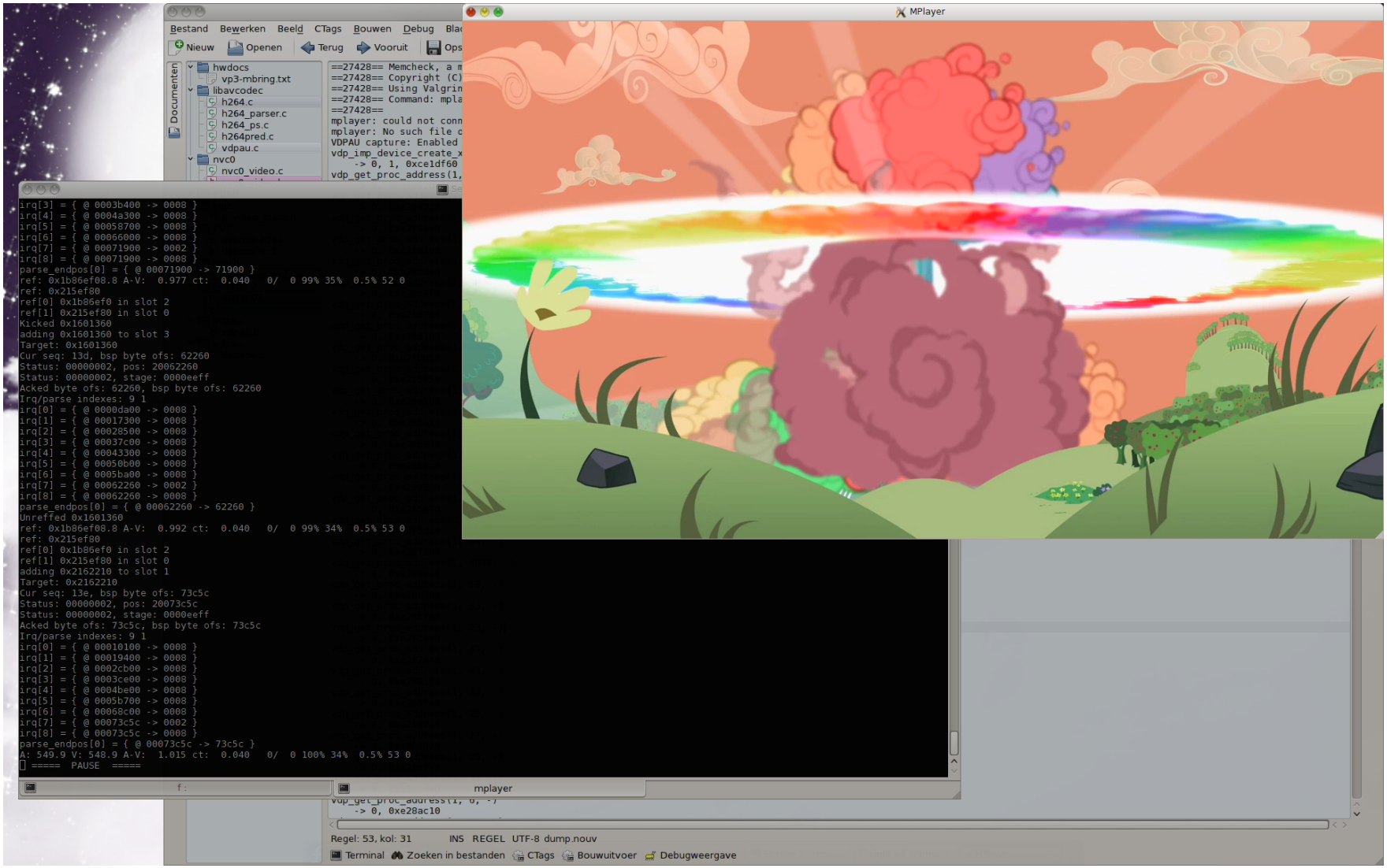
PMPEG

- blob doesn't use this engine
- decodes mpeg1/2

PBSP/PVP/PPPP

- used by nvidia
- decodes every kind of video stream

Components


MPlayer
Bestand Bewerken Beeld CTags Bouwen Debug
Nieuw Openen Terug Vooruit
Documenten
hwdocs
vp3-mbring.txt
libavcodec
h264.c
h264_parser.c
h264_ps.c
h264pred.c
vdpau.c
nvc0
nvc0_video.c
mplayer
Regel: 53, kol: 31 INS REGEL UTF-8 dump.nouv
Terminal Zoeken in bestanden CTags Bouwuitvoer Debugweergave

Summary

Why Nouveau?

- support old cards unsupported by the blob ($<$ nv40)
- support new features (KMS, Xrandr, wayland, d3d1x)
- plug & play support, no need to install and maintain the blob
- for the fun!

Summary

MPEG1/2 video decoding

- using MPlayer
- Kernel-side: done
- Mesa: merged in 8.0

reclocking and opengl

- performance improvements in OpenArena/Nexuiz
- performance improvements in xvmc
- dynamic reclocking (load-based reclocking)!