The Linux 3Dfx HOWTO

Table of Contents

Table of Contents

Table of Contents

The Linux 3Dfx HOWTO

Bernd Kreimeier (bk@gamers.org)

v1.16, 6 February 1998

This document describes 3Dfx graphics accelerator chip support for Linux. It lists some supported hardware, describes how to configure the drivers, and answers frequently asked questions.

4.Answers To Frequently Asked Questions

5.FAQ: Requirements?

6.FAQ: Voodoo Graphics (tm)? 3Dfx?

7.FAQ: Glide? TexUS?

8.FAQ: Glide and XFree86?

9.FAQ: OpenGL/Mesa?

10.FAQ: But Quake?

11.FAQ: Troubleshooting?

1.Introduction

This is the Linux 3Dfx HOWTO document. It is intended as a quick reference covering everything you need to know to install and configure 3Dfx support under Linux. Frequently asked questions regarding the 3Dfx support are answered, and references are given to some other sources of information on a variety of topics related to computer generated, hardware accelerated 3D graphics.

This information is only valid for Linux on the Intel platform. Some information may be applicable to other processor architectures, but I have no first hand experience or information on this. It is only applicable to boards based on 3Dfx technology, any other graphics accelerator hardware is beyond the scope of this document.

1.1 Contributors and Contacts

This document would not have been possible without all the information contributed by other people − those involved in the Linux Glide port and the beta testing process, in the development of Mesa and the Mesa Voodoo drivers, or rewieving the document on behalf of 3Dfx and Quantum3D. Some of them contributed entire sections to this document.

Daryll Strauss daryll@harlot.rb.ca.us did the port, Paul J. Metzger pjm@rbd.com modified the Mesa Voodoo driver (written by David Bucciarelli tech.hmw@plus.it) for Linux, Brian Paul brianp@RA.AVID.COM integrated it with his famous Mesa library. With respect to Voodoo Graphics (tm) accelerated Mesa, additional thanks has to go to Henri Fousse, Gary McTaggart, and the maintainer of the 3Dfx Mesa for DOS, Charlie Wallace Charlie.Wallace@unistudios.com. The folks at 3Dfx, notably Gary

Sanders, Rod Hughes, and Marty Franz, provided valuable input, as did Ross Q. Smith of Quantum3D. The pages on the Voodoo Extreme and Operation 3Dfx websites provided useful info as well, and in some case I relied on the 3Dfx local Newsgroups. The Linux glQuake2 port that uses Linux Glide and Mesa is maintained by Dave Kirsch zoid@idsoftware.com. Thanks to all those who sent e−mail regarding corrections and updates, and special thanks to Mark Atkinson for reminding me of the dual cable setup.

Thanks to the SGML−Tools package (formerly known as Linuxdoc−SGML), this HOWTO is available in several formats, all generated from a common source file. For information on SGML−Tools see its homepage at pobox.com/~cg/sgmltools.

1.2 Acknowledgments

3Dfx, the 3Dfx Interactive logo, Voodoo Graphics (tm), and Voodoo Rush (tm) are registered trademarks of 3Dfx Interactive, Inc. Glide, TexUS, Pixelfx and Texelfx are trademarks of 3Dfx Interactive, Inc. OpenGL is a registered trademark of Silicon Graphics. Obsidian is a trademark of Quantum3D. Other product names are trademarks of the respective holders, and are hereby considered properly acknowledged.

1.3 Revision History

Version 1.03

First version for public release.

Version 1.16

Current version v1.16 6 February 1998.

1.4 New versions of this document

You will find the most recent version of this document at www.gamers.org/dEngine/xf3D/.

New versions of this document will be periodically posted to the comp.os.linux.answers newsgroup. They will also be uploaded to various anonymous ftp sites that archive such information including ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/.

Hypertext versions of this and other Linux HOWTOs are available on many World−Wide−Web sites, including sunsite.unc.edu/LDP/. Most Linux CD−ROM distributions include the HOWTOs, often under the `/usr/doc/`directory, and you can also buy printed copies from several vendors.

If you make a translation of this document into another language, let me know and I'll include a reference to it here.

1.5 Feedback

I rely on you, the reader, to make this HOWTO useful. If you have any suggestions, corrections, or comments, please send them to me (bk@gamers.org), and I will try to incorporate them in the next revision. Please add `HOWTO 3Dfx` to the Subject−line of the mail, so procmail will dump it in the appropriate folder.

Before sending bug reports or questions, *please read all of the information in this HOWTO*, and *send detailed information about the problem*.

If you publish this document on a CD−ROM or in hardcopy form, a complimentary copy would be appreciated. Mail me for my postal address. Also consider making a donation to the Linux Documentation Project to help support free documentation for Linux. Contact the Linux HOWTO coordinator, Tim Bynum (linux−howto@sunsite.unc.edu), for more information.

1.6 Distribution Policy

2.Graphics Accelerator Technology

2.1 Basics

This section gives a *very* cursory overview of computer graphics accelerator technology, in order to help you understand the concepts used later in the document. You should consult e.g. a book on OpenGL in order to learn more.

2.2 Hardware configuration

Graphics accelerators come in different flavors: either as a separate PCI board that is able to pass through the video signal of a (possibly 2D or video accelerated) VGA board, or as a PCI board that does both VGA and 3D graphics (effectively replacing older VGA controllers). The 3Dfx boards based on the Voodoo Graphics (tm) belong to the former category. We will get into this again later.

If there is no address conflict, any 3D accelerator board could be present under Linux without interfering, but in order to access the accelerator, you will need a driver. A combined 2D/3D accelerator might behave differently.

2.3 A bit of Voodoo Graphics (tm) architecture

Usually, accessing texture memory and frame/depth buffer is a major bottleneck. For each pixel on the screen, there are at least one (nearest), four (bi−linear), or eight (tri−linear mipmapped) read accesses to texture memory, plus a read/write to the depth buffer, and a read/write to frame buffer memory.

The Voodoo Graphics (tm) architecture separates texture memory from frame/depth buffer memory by introducing two separate rendering stages, with two corresponding units (Pixelfx and Texelfx), each having a separate memory interface to dedicated memory. This gives an above−average fill rate, paid for restrictions in memory management (e.g. unused framebuffer memory can not be used for texture caching).

Moreover, a Voodoo Graphics (tm) could use two TMU's (texture management or texelfx units), and finally, two Voodoo Graphics (tm) could be combined with a mechanism called Scan−Line Interleaving (SLI). SLI essentially means that each Pixelfx unit effectively provides only every other scanline, which decreases bandwidth impact on each Pixelfx' framebuffer memory.

3.Installation

Configuring Linux to support 3Dfx accelerators involves the following steps:

1. Installing the board.
2. Installing the Glide distribution.
3. Compiling, linking and/or running the application.

The next sections will cover each of these steps in detail.

3.1 Installing the board

Follow the manufacturer's instructions for installing the hardware or have your dealer perform the installation. It should not be necessary to select settings for IRQ, DMA channel, either Plug&Pray (tm) or factory defaults should work. The add–on boards described here are memory mapped devices and do not use IRQ's. The only kind of conflict to avoid is memory overlap with other devices.

As 3Dfx does not develop or sell any boards, do not contact them on any problems.

Troubleshooting the hardware installation

To check the installation and the memory mapping, do `cat /proc/pci`. The output should contain something like

```
  Bus  0, device  12, function  0:
    VGA compatible controller: S3 Inc. Vision 968 (rev 0).
      Medium devsel.  IRQ 11.
      Non-prefetchable 32 bit memory at 0xf4000000.

  Bus  0, device   9, function  0:
    Multimedia video controller: Unknown vendor Unknown device (rev 2).
      Vendor id=121a. Device id=1.
      Fast devsel.  Fast back-to-back capable.
      Prefetchable 32 bit memory at 0xfb000000.
```

for a Diamond Monster 3D used with a Diamond Stealth–64. Additionally a `cat /proc/cpuinfo /proc/meminfo` might be helpfull for tracking down conflicts and/or submitting a bug report.

With current kernels, you will probably get a boot warning like

```
Jun 12 12:31:52 hal kernel: Warning : Unknown PCI device (121a:1).
Please read include/linux/pci.h
```

which could be safely ignored. If you happen to have a board not very common, or have encountered a new revision, you should take the time to follow the advice in `/usr/include/linux/pci.h` and send all necessary information to linux–pcisupport@cao–vlsi.ibp.fr.

If you experience any problems with the board, you should try to verify that DOS and/or Win95 or NT support works. You will probably not receive any useful response from a board manufacturer on a bug report or request regarding Linux. Having dealt with the Diamond support e–mail system, I would not expect useful responses for other operating systems either.

Configuring the kernel

There is no kernel configuration necessary, as long as PCI support is enabled. The Linux Kernel HOWTO should be consulted for the details of building a kernel.

Configuring devices

The current drivers do not (yet) require any special devices. This is different from other driver developments (e.g. the sound drivers, where you will find a `/dev/dsp` and `/dev/audio`). The driver uses the `/dev/mem` device which should always be available. In consequence, you need to use `setuid` or root privileges to access the accelerator board.

3.2 Setting up the Displays

There are two possible setups with add−on boards. You could either pass−through the video signal from your regular VGA board via the accelerator board to the display, or you could use two displays at the same time. Rely to the manual provided by the board manufacturer for details. Both configurations have been tried with the Monster 3D board.

Single screen display solution

This configuration allows you to check basic operations of the accelerator board − if the video signal is not transmitted to the display, hardware failure is possible.

Beware that the video output signal might deteoriate significantly if passed through the video board. To a degree, this is inevitable. However, reviews have complained about below−average of the cables provided e.g. with the Monster 3D, and judging from the one I tested, this has not changed.

There are other pitfalls in single screen configurations. Switching from the VGA display mode to the accelerated display mode will change resolution and refresh rate as well, even if you are using 640x480 e.g. with X11, too. Moreover, if you are running X11, your application is responsible for demanding all keyboard and mouse events, or you might get stuck because of changed scope and exposure on the X11 display (that is effectively invisible when the accelerated mode is used) You could use SVGA console mode instead of X11.

If you are going to use a single screen configuration and switch modes often, remember that your monitor hardware might not enjoy this kind of use.

Single screen dual cable setup

Some high end monitors (e.g. the EIZO F−784−T) come with two connectors, one with 5 BNC connectors for RGB, HSync, VSync, the other e.g. a regular VGA or a 13W3 Sub−D VGA. These displays usually also feature a front panel input selector to safely switch from one to the other. It is thus possible to use e.g. a VGA−to−BNC cable with your high end 2D card, and a VGA−to−13W3 Sub−D cable with your 3Dfx, and effectively run dual screen on one display.

Dual screen display solution

The accelerator board does not need the VGA input signal. Instead of routing the common video output through the accelerator board, you could attach a second monitor to its output, and use both at the same time. This solution is more expensive, but gives best results, as your main display will still be hires and without the signal quality losses involved in a pass−through solution. In addition, you could use X11 and the accelerated full screen display in parallel, for development and debugging.

A common problem is that the accelerator board will not provide any video signal when not used. In consequence, each time the graphics application terminates, the hardware screensave/powersave might kick in depending on your monitors configuration. Again, your hardware might not enjoy being treated like this. You should use

```
setenv SST_DUALSCREEN 1
```

to force continued video output in this setup.

3.3 Installing the Glide distribution

The Glide driver and library are provided as a single compressed archive. Use `tar` and `gzip` to unpack, and follow the instructions in the README and INSTALL accompanying the distribution. Read the install script and run it. Installation puts everything in /usr/local/glide/include,lib,bin and sets the ld.conf to look there. Where it installs and setting ld.conf are independent actions. If you skip the ld.conf step then you need the LD_LIBRARY_PATH.

You will need to install the header files in a location available at compile time, if you want to compile your own graphics applications. If you do not want to use the installation as above (i.e. you insist on a different location), make sure that any application could access the shared libary at runtime, or you will get a response like `can't load library 'libglide.so'`.

Using the detect program

There is a `bin/detect` program in the distribution (the source is not available). You have to run it as root, and you will get something like

```
slot  vendorId   devId   baseAddr0  command  description
----  --------  ------  ----------  -------  -----------
  00    0x8086  0x122d  0x00000000   0x0006  Intel:430FX (Triton)
  07    0x8086  0x122e  0x00000000   0x0007  Intel:ISA bridge
  09    0x121a  0x0001  0xfb000008   0x0002  3Dfx:video multimedia adapter
  10    0x1000  0x0001  0x0000e401   0x0007  ???:SCSI bus controller
  11    0x9004  0x8178  0x0000e001   0x0017  Adaptec:SCSI bus controller
  12    0x5333  0x88f0  0xf4000000   0x0083  S3:VGA-compatible display co
```

as a result. If you do not have root privileges, the program will bail out with

```
Permission denied: Failed to change I/O privilege. Are you root?
```

output might come handy for a bug report as well.

Using the test programs

Within the Glide distribution, you will find a folder with test programs. Note that these test programs are under 3Dfx copyright, and are legally available for use only if you have purchased a board with a 3Dfx chipset. See the LICENSE file in the distribution, or their web site www.3dfx.com for details.

It is recommend to compile and link the test programs even if there happen to be binaries in the distribution. Note that some of the programs will requires some files like `alpha.3df` from the distribution to be available in the same folder. All test programs use the 640x480 screen resolution. Some will request a veriety of single character inputs, others will just state `Press A Key To Begin Test`. Beware of loss of input scope if running X11 on the same screen at the same time.

See the README.test for a list of programs, and other details.

4.Answers To Frequently Asked Questions

The following section answers some of the questions that (will) have been asked on the Usenet news groups and mailing lists. The FAQ has been subdivided into several parts for convenience, namely

- FAQ: Requirements?
- FAQ: Voodoo Graphics (tm)? 3Dfx?
- FAQ: Glide?
- FAQ: Glide and SVGA?
- FAQ: Glide and XFree86?
- FAQ: Glide versus OpenGL/Mesa?
- FAQ: But Quake?
- FAQ: Troubleshooting?

Each section lists several questions and answers, which will hopefully address most problems.

5.FAQ: Requirements?

5.1 What are the system requirements?

A Linux PC, PCI 2.1 compliant, a monitor capable of 640x480, and a 3D accelerator board based on the 3Dfx Voodoo Graphics (tm). It will work on a P5 or P6, with or without MMX. The current version does not use MMX, but it has some optimized code paths for P6.

At one point, some 3Dfx statements seemed to imply that using Linux Glide required using a RedHat distribution. Note that while Linux Glide has originally been ported in a RedHat 4.1 environment, it has been used and tested with many other Linux distributions, including homebrew, Slackware, and Debian 1.3.1.

5.2 Does it work with Linux−Alpha?

There is currently no Linux Glide distribution available for any platform besides i586. As the Glide sources are not available for distribution, you will have to wait for the binary. Quantum3D has DEC Alpha support announced for 2H97. Please contact Daryll Strauss if you are interested in supporting this.

There is also the issue of porting the the assembly modules. While there are alternative C paths in the code, the assembly module in Glide (essentially triangle setup) offered significant performance gains depending on the P5 CPU used.

5.3 Which 3Dfx chipsets are supported?

Currently, the 3Dfx Voodoo Graphics (tm) chipset is supported under Linux. The Voodoo Rush (tm) chipset is not yet supported.

5.4 Is the Voodoo Rush (tm) supported?

The current port of Glide to Linux does not support the Voodoo Rush (tm). An update is in the works.

The problem is that at one point the Voodoo Rush (tm) driver code in Glide depended on Direct Draw. There was an SST96 based DOS portion in the library that could theoretically be used for Linux, as soon as all portions residing in the 2D/Direct Draw/D3D combo driver are replaced.

Thus Voodoo Rush (tm) based boards like the *Hercules Stingray 128/3D* or *Intergraph Intense Rush* are not supported yet.

5.5 Which boards are supported?

There are no officially supported boards, as 3Dfx does not sell any boards. This section does not attempt to list all boards, it will just give an overview, and will list only boards that have been found to cause trouble.

It is important to recognize that Linux support for a given board does not only require a driver for the 3D accelerator component. If a board features its own VGA core as well, support by either Linux SVGA or XFree86 is required as well (see section about Voodoo Rush (tm) chipset). Currently, an add−on solution is recommended, as it allows you to choose a regular graphics board well supported for Linux. There are other aspects discussed below.

All Quantum3D Obsidian boards, independend of texture memory, frame buffer memory, number of Pixelfx and Texelfx units, and SLI should work. Same for all other Voodoo Graphics (tm) based boards, like Orchid Righteous 3D, Canopus Pure 3D, Flash 3D, and Diamond Monster 3D. Voodoo Rush (tm) based boards are not yet supported.

Boards that are not based on 3Dfx chipsets (e.g. manufactured by S3, Matrox, 3Dlabs, Videologic) do *not* work with the 3Dfx drivers and are beyond the scope of this document.

5.6 How do boards differ?

As the board manufacturers are using the same chipset, any differences are due to board design. Examples are quality of the pass−through cable and connectors (reportedly, Orchid provided better quality than Diamond), availability of a TV−compliant video signal output (Canopus Pure 3D), and, most notably, memory size on board.

Most common were boards for games with 2MB texture cache and 2 MB framebuffer memory, however, the Canopus Pure3D comes with a maximal 4 MB texture cache, which is an advantage e.g. with games using dynamically changed textures, and/or illumation textures (Quake, most notably). The memory architecture of a typical Voodoo Graphics (tm) board is described below, in a separate section.

Quantum 3D offers the widest selection of 3Dfx−based boards, and is probably the place to go if you are looking for a high end Voodoo Graphics (tm) based board configuration. Quantum 3D is addressing the visual simulation market, while most of the other vendors are only targetting the consumer−level PC−game market.

5.7 What about AGP?

There is no Voodoo Graphics (tm) or Voodoo Rush (tm) AGP board that I am aware of. I am not aware of AGP support under Linux, and I do not know whether upcmong AGP boards using 3Dfx technology might possibly be supported with Linux.

6.FAQ: Voodoo Graphics (tm)? 3Dfx?

6.1 Who is 3Dfx?

3Dfx is a San Jose based manufacturer of 3D graphics accelerator hardware for arcade games, game consoles, and PC boards. Their official website is www.3dfx.com. 3Dfx does not sell any boards, but other companies do, e.g. Quantum3D.

6.2 Who is Quantum3D?

Quantum3D started as a 3Dfx spin−off, manufacturing high end accelerator boards based on 3Dfx chip technology for consumer and business market, and supplying arcade game technology. See their home page at www.quantum3d.com for additional information. For general inquiries regarding Quantum3D, please send mail to info@quantum3d.

6.3 What is the Voodoo Graphics (tm)?

The Voodoo Graphics (tm) is a chipset manufactured by 3Dfx. It is used in hardware acceleration boards for the PC. See the HOWTO section on supported hardware.

6.4 What is the Voodoo Rush (tm)?

The Voodoo Rush (tm) is a derivate of the Voodoo Graphics (tm) that has an interface to cooperate with a 2D VGA video accelerator, effectively supporting accelerated graphics in windows. This combo is currently not supported with Linux.

6.5 What is the Voodoo 2 (tm)?

The Voodoo 2 (tm) is the successor of the Voodoo Graphics (tm) chipset, featuring several improvements. It is announced for late March 1998, and annoucements of Voodoo 2 (tm) based boards have been published e.g. by Quantum 3D, by Creative Labs, Orchid Technologies, and Diamond Multimedia.

The Voodoo 2 (tm) is supposed to be backwards compatible. However, a new version of Glide will have to be ported to Linux.

6.6 What is VGA pass−though?

The Voodoo Graphics (tm) (but not the Voodoo Rush (tm)) boards are add−on boards, meant to be used with a regular 2D VGA video accelerator board. In short, the video output of your regular VGA board is used as input for the Voodoo Graphics (tm) based add−on board, which by default passes it through to the display also connected to the Voodoo Graphics (tm) board. If the Voodoo Graphics (tm) is used (e.g. by a game), it will disconnect the VGA input signal, switch the display to a 640x480 fullscreen mode with the refresh rate configured by SST variables and the application/driver, and generate the video signal itself. The VGA doesn't need to be aware of this, and won't be.

This setup has several advantages: free choice of 2D VGA board, which is an issue with Linux, as XFree86 drivers aren't available for all chipsets and revisions, and a cost effective migration path to accelerated 3D graphics. It also has several disadvantages: an application using the Voodoo Graphics (tm) might not re−enable video output when crashing, and regular VGA video signal deteoriates in the the pass−through process.

6.7 What is Texelfx or TMU?

Voodoo Graphics (tm) chipsets have two units. The first one interfaces the texture memory on the board, does the texture mapping, and ultimately generates the input for the second unit that interfaces the framebuffer. This one is called Texelfx, aka Texture Management Unit, aka TMU. The neat thing about this is that a board can use two Texelfx instead of only one, like some of the Quantum3D Obsidian boards did, effectively doubling the processing power in some cases, depending on the application.

As each Texelfx can address 4MB texture memory, a dual Texelfx setup has an effective texture cache of up to 8MB. This can be true even if only one Texelfx is actually needed by a particular application, as textures can be distributed to both Texelfx, which are used depending on the requested texture. Both Texelfx are used together to perform certain operations as trilinear filtering and illumination texture/lightmap passes (e.g. in glQuake) in a single pass instead of the two passes that are required with only one Texelfx. To actually exploit the theoretically available speedup and cache size increase, a Glide application has to use both Texelfx properly.

The two Texelfx can not be used separately to each draw a textured triangle at the same time. A triangle is always drawn using whatever the current setup is, which can be to use both Texelfx for a single pass operation combining two textures, or one Texelfx for only a single texture. Each Texelfx can only access its own memory.

6.8 What is a Pixelfx unit?

Voodoo Graphics (tm) chipsets have two units. The second one interfaces the framebuffer and ultimately generates the depth buffer and pixel color updates. This one is called Pixelfx. The neat thing here is that two Pixelfx units can cooperate in SLI mode, like with some of the Quantum3D Obsidian boards, effectively doubling the frame rate.

6.9 What is SLI mode?

SLI means "Scanline Interleave". In this mode, two Pixelfx are connected and render in alternate turns, one handling odd, the other handling even scanlines of the actual output. Inthis mode, each Pixelfx stores only half of the image and half of the depth buffer data in its own local framebuffer, effectively doubling the number of pixels.

The Pixelfx in question can be on the same board, or on two boards properly connected. Some Quantum3D Obsidian boards support SLI with Voodoo Graphics (tm).

As two cards can decode the same PCI addresses and receive the same data, there is not necessarily additional bus bandwidth required by SLI. On the other hand, texture data will have to be replicated on both boards, thus the amount of texture memory effectively stays the same.

6.10 Is there a single board SLI setup?

There are now two types of Quantum3D SLI boards. The intial setup used two boards, two PCI slots, and an interconnect (e.g. the Obsidian 100−4440). The later revision which performs identically is contained on one full−length PCI board (e.g. Obsidian 100−4440SB). Thus a single board SLI solution is possible, and has been done.

6.11 How much memory? How many buffers?

The most essential difference between different boards using the Voodoo Graphics (tm) chipset is the amount and organization of memory. Quantum3D used a three digit scheme to descibe boards. Here is a slightly modifed one (anticipating Voodoo 2 (tm)). Note that if you use more than one Texelfx, they need the same amount of texture cache memory each, and if you combine two Pixelfx, each needs the same amount of frame buffer memory.

```
    "SLI / Pixelfx / Texelfx1 / Texelfx2 "
```

It means that a common 2MB+2MB board would be a `1/2/2/0` solution, with the minimally required total 4Mb of memory. A Canopus Pure 3D would be `1/2/4/0`, or 6MB. An Obsidian−2220 board with two Texelfx would be `1/2/2/2`, and an Obsidian SLI−2440 board would be `2/2/4/4`. A fully featured dual board solution (2 Pixelfx, each with 2 Texelfx and 4MB frame buffer, each Texelfx 4 MB texture cache) would be `2/4/4/4`, and the total amount of memory would be `SLI*(Pixelfx+Texelfx1+Texelfx2)`, or 24 MB.

So there.

6.12 Does the Voodoo Graphics (tm) do 24 or 32 bit color?

No. The Voodoo Graphics (tm) architecture uses 16bpp internally. This is true for Voodoo Graphics (tm), Voodoo Rush (tm) and Voodoo 2 (tm) alike. Quantum3D claims to implement 22−bpp effective color depth with an enhanced 16−bpp frame buffer, though.

6.13 Does the Voodoo Graphics (tm) store 24 or 32 bit z−buffer per pixel?

No. The Voodoo Graphics (tm) architecture uses 16bpp internally for the depth buffer, too. This again is true for Voodoo Graphics (tm), Voodoo Rush (tm) and Voodoo 2 (tm) alike. Again, Quantum3D claims that using the floating point 16−bits per pixel (bpp) depth buffering provides 22−bpp effective Z−buffer precision.

6.14 What resolutions does the Voodoo Graphics (tm) support?

The Voodoo Graphics (tm) chipset supports up to 4 MB frame buffer memory. Presuming double buffering and a depth buffer, a 2MB framebuffer will support a resolution of 640x480. With 4 MB frame buffer, 800x600 is possible.

Unfortunately 960x720 is not supported. The Voodoo Graphics (tm) chipset requires that the amount of memory for a particular resolution must be such that the vertical and horizontal resolutions must be evenly divisible by 32. The video refresh controller, though can output any particular resolution, but the "virtual" size required for the memory footprint must be in dimensions evenly divisible by 32. So, 960x720 actually requires 960x736 amount of memory, and 960x736x2x3 = 4.04MBytes.

However, using two boards with SLI, or a dual Pixelfx SLI board means that each framebuffer will only have to store half of the image. Thus 2 times 4 MB in SLI mode are good up to 1024x768, which is the maximum because of the overall hardware design. You will be able to do 1024x768 tripled buffered with Z, but you will not be able to do e.g. 1280x960 with double buffering.

Note that triple buffering (no VSync synchonization required by the application), stereo buffering (for interfacing LCD shutters) and other more demanding setups will severely decrease the available resolution.

6.15 What texture sizes are supported?

The maximum texture size for the Voodoo Graphics (tm) chipset is 256x256, and you have to use powers of two. Note that for really small textures (e.g. 16x16) you are better off merging them into a large texture, and adjusting your effective texture coordinates appropriately.

6.16 Does the Voodoo Graphics (tm) support paletted textures?

The Voodoo Graphics (tm) hardware and Glide support the palette extension to OpenGL. The most recent version of Mesa does support the `GL_EXT_paletted_texture` and `GL_EXT_shared_texture_palette` extensions.

6.17 What about overclocking?

If you want to put aside considerations about warranty and overheating, and want to do overclocking to boost up performance even further, there is related info out on the web. The basic mechanism is to use Glide environment variables to adjust the clock.

Note that the actual recommended clock is board dependend. While the default clock speed is 50 Mhz, the Diamond Monster 3D property sheet lets you set up a clock of 57 MHz. It all comes down to the design of a specific board, and which components are used with the Voodoo Graphics (tm) chipset – most notably access speed of the RAM in question. If you exceed the limits of your hardware, rendering artifacts will occur to say the least. Reportedly, 57 MHz usually works, while 60 MHz or more is already pushing it.

Increasing the clock frequency also means increasing the waste heat disposed in the chips, in a nonlinear dependency (10% increase in frequency means a lot larger increase in heating). In consequence, for permanent overclocking you might want to educate yourself about ways to add cooling fans to the board in a way that does not affect warranty. A very recommendable source is the "3Dfx Voodoo Heat Report" by Eric van Ballegoie, available on the web.

6.18 Where could I get additional info on Voodoo Graphics (tm)?

There is a FAQ by 3Dfx, which should be available at their web site. You will find retail information at the following locations: www.3dfx.com and www.quantum3d.com.

Inofficial sites that have good info are "Voodoo Extreme" at www.ve3d.com, and "Operation 3Dfx" at www.ve3d.com.

7.FAQ: Glide? TexUS?

7.1 What is Glide anyway?

Glide is a proprietary API plus drivers to access 3D graphics accelerator hardware based on chipsets manufactured by 3Dfx. Glide has been developed and implemented for DOS, Windows, and Macintosh, and has been ported to Linux by Daryll Strauss.

7.2 What is TexUS?

In the distribution is a `libtexus.so`, which is the 3Dfx Interactive Texture Utility Software. It is an image processing libary and utility program for preparing images for use with the 3Dfx Interactive Glide library. Features of TexUS include file format conversion, MIPmap creation, and support for 3Dfx Interactive Narrow Channel Compression textures.

The TexUS utility program `texus` reads images in several popular formats (TGA, PPM, RGT), generates MIPmaps, and writes the images as 3Dfx Interactive textures files (see e.g. alpha.3df, as found in the distribution) or as an image file for inspection. For details on the parameters for `texus`, and the API, see the TexUS documentation.

7.3 Is Glide freeware?

Nope. Glide is neither GPL'ed nor subject to any other public license. See LICENSE in the distribution for any details. Effectively, by downloading and using it, you agree to the End User License Agreement (EULA) on the 3Dfx web site. Glide is provided as binary only, and you should neither use nor distribute any files but the ones released to the public, if you have not signed an NDA. The Glide distribution including the test program sources are copyrighted by 3Dfx.

The same is true for all the sources in the Glide distribution. In the words of 3Dfx: These are not public domain, but they can be freely distributed to owners of 3Dfx products only. No card, No code!

7.4 Where do I get Glide?

The entire 3Dfx SDK is available for download off their public web−site located at www.3dfx.com/software/download_glide.html. Anything else 3Dfx publicly released by 3Dfx is nearby on their website, too.

There is also an FTP site, ftp.3dfx.com. The FTP has a longer timeout, and some of the larger files have been broken into 3 files (approx. 3MB each).

7.5 Is the Glide source available?

Nope. The Glide source is made available only based on a special agreement and NDA with 3Dfx.

7.6 Is Linux Glide supported?

Currently, Linux Glide is unsupported. Basically, it is provided under the same disclaimers as the 3Dfx GL DLL (see below).

However, 3Dfx definitely wants to provide as much support as possible, and is in the process of setting up some prerequisites. For the time being, you will have to rely on the 3Dfx newsgroup (see below).

In addition, the Quantum3D web page claims that Linux support (for Obsidian) is planned for both Intel and AXP architecture systems in 2H97.

7.7 Where could I post Glide questions?

There are newsgroups currently available only on the NNTP server news.3dfx.com run by 3Dfx. This USENET groups are dedicated to 3Dfx and Glide in general, and will mainly provide assistance for DOS, Win95, and NT. The current list includes:

```
3dfx.events
3dfx.games.glquake
3dfx.glide
3dfx.glide.linux
3dfx.products
3dfx.test
```

---

and the `3dfx.oem.products.*` group for specific boards, eg. `3dfx.oem.products.quantum3d.obsidian`. Please use news.3dfx.com/3dfx.glide.linux for all Lnux Glide related questions.

A mailing list dedicated to Linux Glide is in preparation for 1Q98. Send mail to majordomo@gamers.org, no subject, body of the message `info linux-3dfx` to get information about the posting guidelines, the hypermail archive and how to subscribe to the list or the digest.

7.8 Where to send bug reports?

Currently, you should rely on the newsgroup (see above), that is news.3dfx.com/3dfx.glide.linux. There is no official support e−mail set up yet. For questions not specific to Linux Glide, make sure to use the other newsgroups.

7.9 Who is maintaining it?

3Dfx will appoint an official maintainer soon. Currently, inofficial maintainer of the Linux Glide port is Daryll Strauss. Please post bug reports in the newsgroup (above). If you are confident that you found a bug not previously reported, please mail to Daryll at daryll@harlot.rb.ca.us

7.10 How can I contribute to Linux Glide?

You could submit precise bug reports. Providing sample programs to be included in the distribution is another possibility. A major contribution would be adding code to the Glide based Mesa Voodoo driver source. See section on Mesa Voodoo below.

7.11 Do I have to use Glide?

Yes. As of now, there is no other Voodoo Graphics (tm) driver available for Linux. At the lowest level, Glide is the only interface that talks directly to the hardware. However, you can write OpenGL code without knowing anything about Glide, and use Mesa with the Glide based Mesa Voodoo driver. It helps to be aware of the involvement of Glide for recognizing driver limitations and bugs, though.

7.12 Should I program using the Glide API?

That depends on the application you are heading for. Glide is a proprietary API that is partly similar to OpenGL or Mesa, partly contains features only available as EXTensions to some OpenGL implementations, and partly contains features not available anywhere but within Glide.

If you want to use the OpenGL API, you will need Mesa (see below). Mesa, namely the Mesa Voodoo driver, offers an API resembling the well documented and widely used OpenGL API. However, the Mesa Voodoo driver is in early alpha, and you will have to accept performance losses and lack of support for some features.

In summary, the decision is up to you − if you are heading for maximum performance while accepting potential problems with porting to non−3Dfx hardware, Glide is not a bad choice. If you care about maintenance, OpenGL might be the best bet in the long run.

7.13 What is the Glide current version?

The current version of Linux Glide is 2.4. The next version will probably be identical to the current version for DOS/Windows, which is 2.4.3, which comes in two distributions. Right now, various parts of Glide are different for Voodoo Rush (tm) (VR) and Voodoo Graphics (tm) (VG) boards. Thus you have to pick up separate distributions (under Windows) for VR and VG. The same will be true for Linux. There will possibly be another chunk of code and another distribution for Voodoo 2 (tm) (V2) boards.

There is also a Glide 3.0 in preparation that will extend the API for use of triangle fans and triangle strips, and provide better state change optimization. Support for fans and strips will in some situations significantly reduce the amount of data sent ber triangle, and the Mesa driver will benefit from this, as the OpenGL API has separate modes for this. For a detailed explanation on this see e.g. the OpenGL documentation.

7.14 Does it support multiple Texelfx already?

Multiple Texelfx/TMU's can be used for single pass trilinear mipmapping for improvement image quality without performance penalty in current Linux Glide already. You will need a board with two Texelfx (that is, one of the appropriate Quantum3D Obsidian boards). The application needs to specify the use of both Texelfx accordingly, it does not happen automatically.

Note that because most applications are implemented for consumer boards with a single Texelfx, they might not query the presence of a second Texelfx, and thus not use it. This is not a flaw of Glide but of the application.

7.15 Is Linux Glide identical to DOS/Windows Glide?

The publicly available version of Linux Glide should be identical to the respective DOS/Windows versions. Delays in releasing the Linux port of newer DOS/Windows releases are possible.

7.16 Where to I get information on Glide?

There is exhaustive information available from 3Dfx. You could download it from their home page at www.3dfx.com/software/download_glide.html. These are for free, presuming you bought a 3Dfx hardware based board. Please read the licensing regulations.

Basically, you should look for some of the following:

- *Glide Release Notes*
- *Glide Programming Guide*
- *Glide Reference Manual*
- *Glide Porting Guide*

- *TexUs Texture Utility Software*
- *ATB Release Notes*
- *Installing and Using the Obsidian*

These are available as Microsoft Word documents, and part of the Windows Glide distribution, i.e. the self−extracting archive file. Postscript copies for separate download should be available at www.3dfx.com as well. Note that the release numbers are not always in sync with those of Glide.

7.17 Where to get some Glide demos?

You will find demo sources for Glide within the distribution (test programs), and on the 3Dfx home page. The problem with the latter is that some require ATB. To port these demos to Linux, the event handling has to be completely rewritten.

In addition, you might find useful some of the OpenGL demo sources accompanying Mesa and GLUT. While the Glide API is different from the OpenGL API, they target the same hardware rendering pipeline.

7.18 What is ATB?

Some of the 3Dfx demo programs for Glide depend not only on Glide but also on 3Dfx's proprietary Arcade Toolbox (ATB), which is available for DOS and Win32, but has not been ported for Linux. If you are a devleoper, the sources are available within the Total Immersion program, so porting ATB to Linux would be possible.

8.FAQ: Glide and XFree86?

8.1 Does it run with XFree86?

Basically, the Voodoo Graphics (tm) hardware does not care about X. The X server will not even notice that the video signal generated by the VGA hardware does not reach the display in single screen configurations. If your application is not written X aware, Glide switching to full screen mode might cause problems (see troubleshooting section). If you do not want the overhead of writing an X11−aware application, you might want to use SVGA console mode instead.

So yes, it does run with XFree86, but no, it is not cooperating if you don't write your application accordingly. You can use the Mesa "window hack", which will be significantly slower than fullscreen, but still a lot faster than software rendering (see section below).

8.2 Does it only run full screen?

See above. The Voodoo Graphics (tm) hardware is not window environment aware, neither is Linux Glide. Again, the experimental Mesa "window hack" covered below will allow for pasting the Voodoo Graphics (tm) board framebuffer's content into an X11 window.

8.3 What is the problem with AT3D/Voodoo Rush (tm) boards?

There is an inherent problem when using Voodoo Rush (tm) boards with Linux: Basically, these boards are meant to be VGA 2D/3D accelerator boards, either as a single board solution, or with a Voodoo Rush (tm) based daughterboard used transparently. The VGA component tied to the Voodoo Rush (tm) is a Alliance Semiconductor's ProMotion−AT3D multimedia accelerator. To use this e.g. with XFree86 at all, you need a driver for the AT3D chipset.

There is a mailing list on this, and a web site with FAQ at www.frozenwave.com/linux−stingray128. Look there for most current info. There is a SuSE maintained driver at ftp.suse.com/suse_update/special/xat3d.tgz. Reportedly, the XFree86 SVGA server also works, supporting 8, 16 and 32 bpp. Official support will probably be in XFree86 4.0. XFree86 decided to prepare an intermediate XFree86 3.3.2 release as well, which might already address the issues.

The following `XF86Config` settings reportedly work.

```
# device section settings
Chipset "AT24"
Videoram 4032

# videomodes tested by Oliver Schaertel
#  25.18  28.32  for 640 x 480   (70hz)
#  61.60         for 1024 x 786  (60hz)
#  120           for 1280 x 1024 (66hz)
```

In summary, there is nothing prohibiting this except for the fact that the drivers in XFree86 are not yet finished.

If you want a more technical explanation: Voodoo Rush (tm) support requires X server changes to support grabbing a buffer area in the video memory on the AT3D board, as the Voodoo Rush (tm) based boards need to store their back buffer and z buffer there. This memory allocation and locking requirement is not a 3Dfx specific problem, it is also needed e.g. for support of TV capture cards, and is thus under active development for XFree86. This means changes at the device dependend X level (thus XAA), which are currently implemented as an extension to XFree86 DGA (Direct Graphics Access, an X11 extension proposal implemented in different ways by Sun and XFree86, that is not part of the final X11R6.1 standard and thus not portable). It might be part of an XFree86 GLX implementation later on. The currently distributed X servers assume they have full control of the framebuffer, and use anything that is not used by the visual region of the framebuffer as pixmap cache, e.g. for caching fonts.

8.4 What about GLX for XFree86?

There are a couple of problems.

The currently supported Voodoo Graphics (tm) hardware and the available revision of Linux Glide are full screen only, and not set up to share a framebuffer with a window environment. Thus GLX or other integration with X11 is not yet possible.

The Voodoo Rush (tm) might be capable of cooperating with XFree86 (that is, an SVGA compliant board will work with the XFree86 SVGA server), but it is not yet supported by Linux Glide, nor do S3 or other XFree86 servers support these boards yet.

In addition, GLX is tied to OpenGL or, in the Linux case, to Mesa. The XFree86 team is currently working on integrating Mesa with their X Server. GLX is in beta, XFree86 3.3 has the hooks for GLX. See Steve Parker's GLX pages at www.cs.utah.edu/~sparker/xfree86−3d/ for the most recent information. Moreover, there is a joint effort by XFree86 and SuSe, which includes a GLX, see www.suse.de/~sim/. Currently, Mesa still uses its GLX emulation with Linux.

8.5 Glide and commerical X Servers?

I have not received any mail regarding use of Glide and/or Mesa with commercial X Servers. I would be interested to get confirmation on this, especially on Mesa and Glide with a commercial X Server that has GLX support.

8.6 Glide and SVGA?

You should have no problems running Glide based applications either single or dual screen using VGA modes. It might be a good idea to set up the 640x480 resolution in the SVGA modes, too, if you are using a single screen setup.

8.7 Glide and GGI?

A GGI driver for Glide is under development by Jon M. Taylor, but has not officially been released and was put on hold till completion of GGI 0.0.9. For information about GGI see synergy.caltech.edu/~ggi/. If you are adventurous, you might find the combination of XGGI (a GGI based X Server for XFree86) and GGI for Glide an interesting prospect. There is also a GGI driver interfacing the OpenGL API; tested with unaccelerated Mesa. Essentially, this means X11R6 running on a Voodoo Graphics (tm), using either Mesa or Glide directly.

9.FAQ: OpenGL/Mesa?

9.1 What is OpenGL?

OpenGL is an immediate mode graphics programming API originally developed by SGI based on their previous proprietary Iris GL, and became in industry standard several years ago. It is defined and maintained by the Architectural Revision Board (ARB), an organization that includes members as SGI, IBM, and DEC, and Microsoft.

OpenGL provides a complete feature set for 2D and 3D graphics operations in a pipelined hardware accelerated architecture for triangle and polygon rendering. In a broader sense, OpenGL is a powerful and generic toolset for hardware assisted computer graphics.

9.2 Where to get additional information on OpenGL?

The official site for OpenGL maintained by the members of the ARB, is www.opengl.org,

A most recommended site is Mark Kilgard's Gateway to OpenGL Info at reality.sgi.com/mjk_asd/opengl–links.html: it provides pointers to book, online manual pages, GLUT, GLE, Mesa, ports to several OS, tons of demos and tools.

If you are interested in game programming using OpenGL, there is the `OpenGL–GameDev–L@fatcity.com` at `Listserv@fatcity.com`. Be warned, this is a high traffic list with very technical content, and you will probably prefer to use `procmail` to handle the 100 messages per day coming in. You cut down bandwidth using the `SET OpenGL–GameDev–L DIGEST` command. It is also not appropriate if you are looking for introductions. The archive is handled by the ListServ software, use the `INDEX OpenGL–GameDev–L` and `GET OpenGL–GameDev–L "filename"` commands to get a preview before subscribing.

9.3 Is Glide an OpenGL implementation?

No, Glide is a proprietary 3Dfx API which several features specific to the Voodoo Graphics (tm) and Voodoo Rush (tm). A 3Dfx OpenGL is in preparation (see below). Several Glide features would require EXTensions to OpenGL, some of which already found in other implementations (e.g. paletted textures).

The closest thing to a hardware accelerated Linux OpenGL you could currently get is Brian Paul's Mesa along with David Bucciarelli's Mesa Voodoo driver (see below).

9.4 Is there an OpenGL driver from 3Dfx?

Both the 3Dfx website and the Quantum3D website announced OpenGL for Voodoo Graphics (tm) to be available 4Q97. The driver is currently in Beta, and accessible only to registered deverloper's under written Beta test agreement.

A linux port has not been announced yet.

9.5 Is there a commercial OpenGL for Linux and 3Dfx?

I am not aware of any third party commercial OpenGL that supports the Voodoo Graphics (tm). Last time I paid attention, neither MetroX nor XInside OpenGL did.

9.6 What is Mesa?

Mesa is a free implementation of the OpenGL API, designed and written by Brian Paul, with contributions from many others. Its performance is competitive, and while it is not officially certified, it is an almost fully compliant OpenGL implementation conforming to the ARB specifications − more complete than some commercial products out, actually.

9.7 Does Mesa work with 3Dfx?

The latest Mesa MesaVer; release works with Linux Glide 2.4. In fact, support was included in earlier versions, however, this driver is still under development, so be prepared for bugs and less than optimal performance. It is steadily improving, though, and bugs are usually fixed very fast.

You will need to get the Mesa library archive from the iris.ssec.wisc.edu FTP site. It is recommended to subscribe to the mailing list as well, especially when trying to track down bugs, hardware, or driver limitations. Make sure to get the most recent distribution. A Mesa−3.0 is in preparation.

9.8 How portable is Mesa with Glide?

It is available for Linux and Win32, and any application based on Mesa will only have the usual system specific code, which should usually mean XWindows vs. Windows, or GLX vs. WGL. If you use e.g. GLUT or Qt, you should get away with any system specifics at all for virtually most applications. There are only a few issues (like sampling relative mouse movement) that are not adressed by the available portable GUI toolkits.

Mesa/Glide is also available for DOS. The port which is 32bit DOS is maintained by Charlie Wallace and kept up to date with the main Mesa base. See www.geocities.com/~charlie_x/.for the most current releases.

9.9 Where to get info on Mesa?

The Mesa home page is at www.ssec.wisc.edu/~brianp/Mesa.html. There is an archive of the Mesa mailing list. at www.iqm.unicamp.br/mesa/. This list is not specific to 3Dfx and Glide, but if you are interested in using 3Dfx hardware to accelerate Mesa, it is a good place to start.

9.10 Where to get information on Mesa Voodoo?

For latest information on the Mesa Voodoo driver maintained by David Bucciarelli tech.hmw@plus.it see the home page at www−hmw.caribel.pisa.it/fxmesa/.

9.11 Does Mesa support multitexturing?

Not yet (as of Mesa 2.6), but it is on the list. In Mesa you will probably have to use the OpenGL `EXT_multitexture` extension once it is available. There is no final specification for multitextures in OpenGL, which is supposed to be part of the upcoming OpenGL 1.2 revision. There might be a Glide driver specific implementation of the extension in upcoming Mesa releases, but as long as only certain Quantum3D Obsidian boards come with multiple TMU's, it is not a top priority. This will surely change once Voodoo 2 (tm) based boards are in widespread use.

9.12 Does Mesa support single pass trilinear mipmapping?

Multiple TMU's should be used for single pass trilinear mipmapping for improvement image quality without performance penalty in current Linux Glide already. Mesa support is not yet done (as of Mesa 2.6), but is in preparation.

9.13 What is the Mesa "Window Hack"?

The most recent revisions of Mesa contain an experimental feature for Linux XFree86. Basically, the GLX emulation used by Mesa copies the contents of the Voodoo Graphics (tm) board's most recently finished framebuffer content into video memory on each `glXSwapBuffers` call. This feature is also available with Mesa for Windows.

This obviously puts some drain on the PCI, doubled by the fact that this uses X11 MIT SHM, not XFree86 DGA to access the video memory. The same approach could theoretically be used with e.g. SVGA. The major benefit is that you could use a Voodoo Graphics (tm) board for accelerated rendering into a window, and that you don't have to use the VGA passthrough mode (video output of the VGA board deteoriates in passing through, which is very visible with high end monitors like e.g. EIZO F784−T).

Note that this experimental feature is *NOT* Voodoo Rush (tm) support by any means. It applies only to the Voodoo Graphics (tm) based boards. Moreover, you need to use a modified GLUT, as interfacing the window management system and handling the events appropriately has to be done by the application, it is not handled in the driver.

Make really sure that you have enabled the following environment variables:

```
export SST_VGA_PASS=1          # to stop video signal switching
export SST_NOSHUTDOWN=1        # to stop video signal switching
export MESA_GLX_FX="window"    # to initiate Mesa window mode
```

If you manage to forget one of the SST variables, your VGA board will be shut off, and you will loose the display (but not the actual X). It is pretty hard to get that back being effectively blind.

Finally, note that the libMesaGL.a (or .so) library can contain multiple client interfaces. I.e. the GLX, OSMesa, and fxMesa (and even SVGAMesa) interfaces call all be compiled into the same libMesaGL.a. The client program can use any of them freely, even simultaneously if it's careful.

9.14 How about GLUT?

Mark Kilgard's GLUT distribution is a very good place to get sample applications plus a lot of useful utilities. You will find it at reality.sgi.com/mjk_asd/glut3/, and you should get it anyway. The current release is GLUT 3.6, and discussion on a GLUT 3.7 (aka GameGLUT) has begun. Note that Mark Kilgard has left SGI recently, so the archive might move some time this year − for the time being it will be kept at SGI.

There is also a GLUT mailing list, `glut@perp.com`. Send mail to majordomo@perp.com, with the (on of the) following in the body of your email message:

```
   help
   info glut
   subscribe glut
   end
```

As GLUT handles double buffers, windows, events, and other operations closely tied to hardware and operating system, using GLUT with Voodoo Graphics (tm) requires support, which is currently in development within GLX for Mesa. It already works for most cases.

10.FAQ: But Quake?

10.1 What about that 3Dfx GL driver for Quake?

The 3Dfx Quake GL, aka mini−driver, aka miniport, aka Game GL, aka 3Dfx GL alpha, implemented only a Quake−specific subset of OpenGL (see http://www.cs.unc.edu/~martin/3dfx.html for an inofficial list of supported code paths). It is not supported, and not updated anymore. It was a Win32 DLL (`opengl32.dll`) released by 3Dfx and was available for Windows only. This DLL is not, and will not be ported to Linux.

10.2 Is there a 3Dfx based glQuake for Linux?

Yes. A Quake linuxquake v0.97 binary has been released based on Mesa with Glide. The Quake2 q2test binary for Linux and Voodoo Graphics (tm) has been made available as well. A full Quake2 for Linux was released in January 1998, with linuxquake2−3.10. Dave "Zoid" Kirsch is the official maintainer of all Linux ports of Quake, Quakeworld, and Quake2, including all the recent Mesa based ports. Note that all Linux ports, including the Mesa based ones, are not officially supported by id Software.

See ftp.idsoftware.com/idstuff/quake/unix/ for the latest releases.

10.3 Does glQuake run in an XFree86 window?

A revision of Mesa and the Mesa−based Linux glQuake is in preparation. Mesa already does support this by GLX, but Linux glQuake does not use GLX.

10.4 Known Linux Quake problems?

Here is an excerpt, as of January 7th, 1998. I omitted most stuff not specific to &3Dfx; hardware.

- You really should run Quake2 as root when using the SVGALib and/or GL renders. You don't have to run as root for the X11 refresh, but the modes on the mouse and sound devices must be read/writable by whatever user you run it as. Dedicated server requires no special permissions.
- X11 has some garbage on the screen when 'loading'. This is normal in 16bit color mode. X11 doesn't

work in 24bit (TrueColor). It would be very slow in any case.

- Some people are experiencing crashes with the GL renderer. Make sure you install the libMesa that comes with Quake2! Older versions of libMesa don't work properly.
- If you are experience video 'lag' in the GL renderer (the frame rate feels like it's lagging behind your mouse movement) type "gl_finish 1" in the console. This forces update on a per frame basis.
- When running the GL renderer, make sure you have killed selection and/or gpm or the mouse won't work as they won't "release" it while Quake2 is running in GL mode.

10.5 Know Linux Quake security problems?

As Dave Kirsch posted on January 28th, 1998: an exploit for Quake2 under Linux has been published. Quake2 is using shared libraries. While the READMRE so far does not specifically mention it, note that Quake2 should not be `setuid`.

If you want to use the `ref_soft` and `ref_gl` renderers, you should run Quake2 as root. Do not make the binary setuid. You can only run both those renderers at the console only, so being root is not that much of an issue.

The X11 render does not need any root permissions (if `/dev/dsp` is writable by others for sound). The dedicated server mode does not need to be root either, obviously.

Problems such as root requirements for games has been sort of a sore spot in Linux for a number of years now. This is one of the goals that e.g. GGI is targetting to fix. A `ref_ggi` might be supported in the near future.

10.6 Does LinuxQuake use multitexturing?

To my understadnding, glQuake will use a multitexture EXTension if the OpenGL driver in question offers it. The current Mesa implementation and the Glide driver for Linux do not yet support this extension, so for the time being the answer is no. See section on Mesa and multitexturing for details.

10.7 Where can I get current information on Linux glQuake?

Try some of these sites: the "The Linux Quake Resource" at linuxquake.telefragged.com, or the "Linux Quake Page" at www.planetquake.com/threewave/linux/. Alternatively, you could look for Linux Quake sites in the "SlipgateCentral" database at www.slipgatecentral.com.

11.FAQ: Troubleshooting?

11.1 Has this hardware been tested?

See hardware requirements list above. I currently do not maintain a conclusive list of vendors and boards, as no particular board specific problems have been verified. Currently, only 3Dfx and Quantum3D provide boards for testing to the developers, so Quantum3D consumer boards are a safe bet. Every other Voodoo Graphics (tm) based board should work, too. I have reports regarding the Orchid Righteous 3D, Guillemot Maxi 3D Gamer, and Diamond Monster 3D.

If you are a board manufacturer who wants to make sure his Voodoo Graphics (tm), Voodoo Rush (tm) or Voodoo 2 (tm) boards work with upcoming releases of Linux, Xfree86, Linux Glide and/or Mesa, please contact me, and I will happily forward your request to the persons maintaining the drivers in question. If you are interested in support for Linux Glide on other then the PC platfrom, e.g. DEC Alpha, please contact the maintainer of Linux Glide Daryll Strauss, at daryll@harlot.rb.ca.us

11.2 Failed to change I/O privilege?

You need to be root, or `setuid` your application to run a Glide based application. For DMA, the driver accesses `/dev/mem`, which is not writeable for anybody but root, with good reasons. See the README in the Glide distribution for Linux.

11.3 Does it work without root privilege?

There are compelling case where the setuid requirement is a problem, obviously. There are currently solutions in preparation, which require changes to the library internals itself.

11.4 Displayed images looks awful (single screen)?

If you are using the analog pass through configuration, the common SVGA or X11 display might look pretty bad. You could try to get a better connector cable than the one provided with the accelerator board (the ones delivered with the Diamond Monster 3D are reportedly worse then the one accompanying the Orchid Righteous 3D), but up to a degree there will inevitably be signal loss with an additional transmission added.

If the 640x480 full screen image created by the accelerator board does look awful, this might indicate a real hardware problem. You will have to contact the board manufacturer, not 3Dfx for details, as the quality of the video signal has nothing to do with the accelerator – the board manufacturer chooses the RAMDAC, output drivers, and other components responsible.

11.5 The last frame is still there (single or dual screen)?

You terminated your application with Ctrl−C, or it did not exit normally. The accelerator board will dutifully provide the current content of the framebuffer as a video signal unless told otherwise.

11.6 Powersave kicks in (dual screen)?

When you application terminates in dual screen setups, the accelerator board does not provide video output any longer. Thus powersave kicks each time. To avoid this, use

```
setenv SST_DUALSCREEN 1
```

11.7 My machine seem to lock (X11, single screen)?

If you are running X when calling a Glide application, you probably moved the mouse out of the window, and the keyboard inputs do not reach the application anymore.

If you application is supposed to run concurrently with X11, it is recommend to expose a full screen window, or use the `XGrabPointer` and `XGrabServer` functions to redirect all inputs to the application while the X server cannot access the display. Note that grabbing all input with `XGrabPointer` and `XGrabServer` does not qualify as well−behaved application, and that your program might block the entire system.

If you experience this problem without running X, be sure that there is no hardware conflict (see below).

11.8 My machine locks (single or dual screen)?

If the system definitely does not respond to any inputs (you are running two displays and know about the loss of focus), you might experience a more or less subtle hardware conflict. See installation troubleshooting section for details.

If there is no obvious address conflict, there might still be other problems (below). If you are writing your own code the most common reason for locking is that you didn't snap your vertices. See the section on snapping in the Glide documentation.

11.9 My machine locks (used with S3 VGA board)?

It is possible you have a problem with memory region overlap specific to S3. There is some info and a patch to the so−called S3 problem in the 3Dfx web site, but these apply to Windows only. To my understanding, the cause of the problem is that some S3 boards (older revisions of Diamond Stealth S3 968) reserve more memory space than actually used, thus the Voodoo Graphics (tm) has to be mapped to a different location. However, this has not been reported as a problem with Linux, and might be Windows−specific.

11.10 No address conflict, but locks anyway?

If you happen to use a motherboard with non−standard or incomplete PCI support, you could try to shuffle the boards a bit. I am running an ASUS TP4XE that has that non−standard modified "Media Slot", i.e. PCI slot4 with additional connector for ASUS−manufactured SCSI/Sound combo boards, and I experienced severe problems while running a Diamond Monster 3D in that slot. The system operates flawlessly since I put the board in one of the regular slots.

11.11 Mesa runs, but does not access the board?

Be sure that you recompiled all the libraries (including the toolkits the demo programs use − remember that GLUT does not yet support Voodoo Graphics (tm)), and that you removed the older libraries, run `ldconfig`, and/or set your `LD_LIBRARY_PATH` properly. Mesa supports several drivers in parallel (you could use X11 SHM, off screen rendering, and Mesa Voodoo at the same time), and you might have to create and switch contexts explicitely (see `MakeCurrent` function) if the Voodoo Graphics (tm) isn't chosen by default.

11.12 Resetting dual board SLI?

If a Quantum 3D Obsidian board using in an SLI setup exits abruptly (i.e., the application crashes, or is aborted by user), the boards are left in an undefined state. With the dual−board set, you can run a program called `resetsli` to reset them. Until you run the `resetsli` program, you will not be able to re−initialize the Obsidian board.

11.13 Resetting single board SLI?

The `resetsli` program mentioned above does not yet work with a single board Obsidian SLI (e.g. the Obsidian 100−4440SB). You will have to reboot your system by reset in order to reset the board.