实验1.1 OpenGL图形编程环境配置

1. 实验目的
2. 掌握g++编译链接的方法。
3. 掌握linux系统内的源码编译方法。
4. 了解和掌握OpenGL的环境配置。
5. 掌握OpenGL工程项目的建立和基本设置。
6. 实验环境
7. 硬件环境：主流配置，最好配备独立显卡。
8. 软件环境：
   1. 操作系统：ubuntu (16及以上)
   2. 构建工具：g++ （也可以使用cmake，参考MacOS下教程的命令）
   3. 开发库：OpenGL, GLFW, GLAD, GLM

后续的实验环境与此相同，以后将不再赘述。

1. 理论背景
2. OpenGL简介

OpenGL（Open Graphics Library）是指定义了一个跨编程语言、跨平台的编程接口规格的专业的图形程序接口。它用于三维图像（二维的亦可），是一个功能强大，调用方便的底层图形库。OpenGL是行业领域中最为广泛接纳的 2D/3D 图形 API，其自诞生至今已催生了各种计算机平台及设备上的数千优秀应用程序。

1. GLFW简介

GLFW（Graphics Library Framework），它是一个小型 C 库，允许使用 OpenGL 上下文创建和管理窗口，从而也可以使用多个监视器和视频模式。它同时提供对键盘、鼠标和操纵杆输入的访问。

1. GLAD简介

由于OpenGL驱动版本众多，大多数函数的位置都无法在编译时确定下来，需要在运行时进行查询，使得开发过程较为繁琐。GLAD (Multi-Language GL/GLES/EGL/GLX/WGL Loader-Generator based on the official specs) 是一个开源的库，可以提供一种更为简单的方法加载所有相关的OpenGL函数。

1. GLM简介

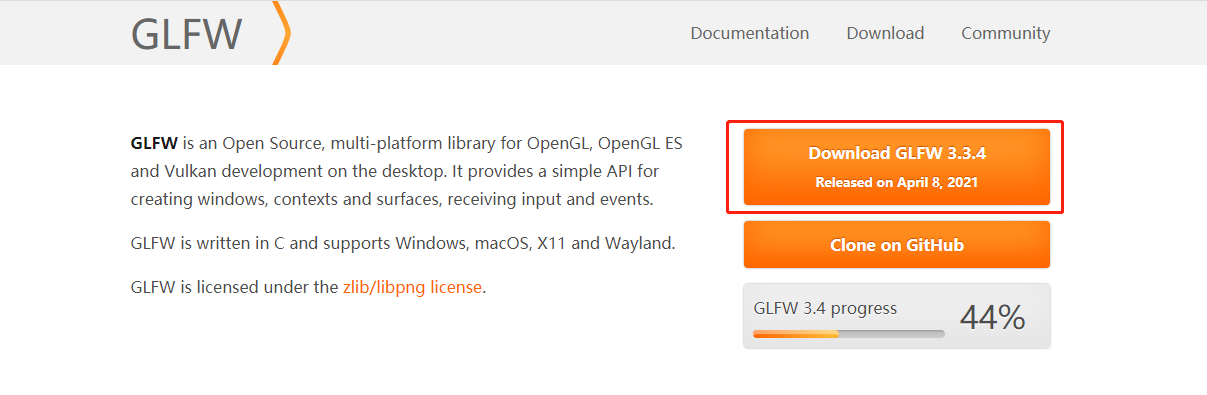
GLM（OpenGL Mathematics）是专门为OpenGL量身定做的数学库，它是一个只有头文件的库，可以快速实现矩阵变换等各种图形学中常用的几何计算。

1. 三者的关系

OpenGL定义标准 → 显卡厂商编写驱动 → GLAD提供接口 → GLFW建立窗口

1. 实验内容
2. **环境配置**

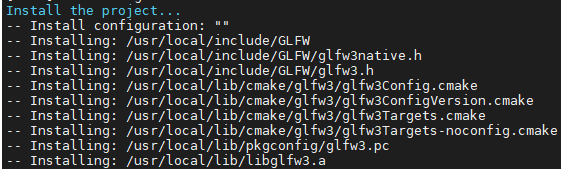
（1）依赖库GLFW的安装

进入GLFW官网（https://www.glfw.org/），下载源文件。

使用unzip glfw-3.3.4.zip -d glfw-3.3.4命令解压该压缩文件。

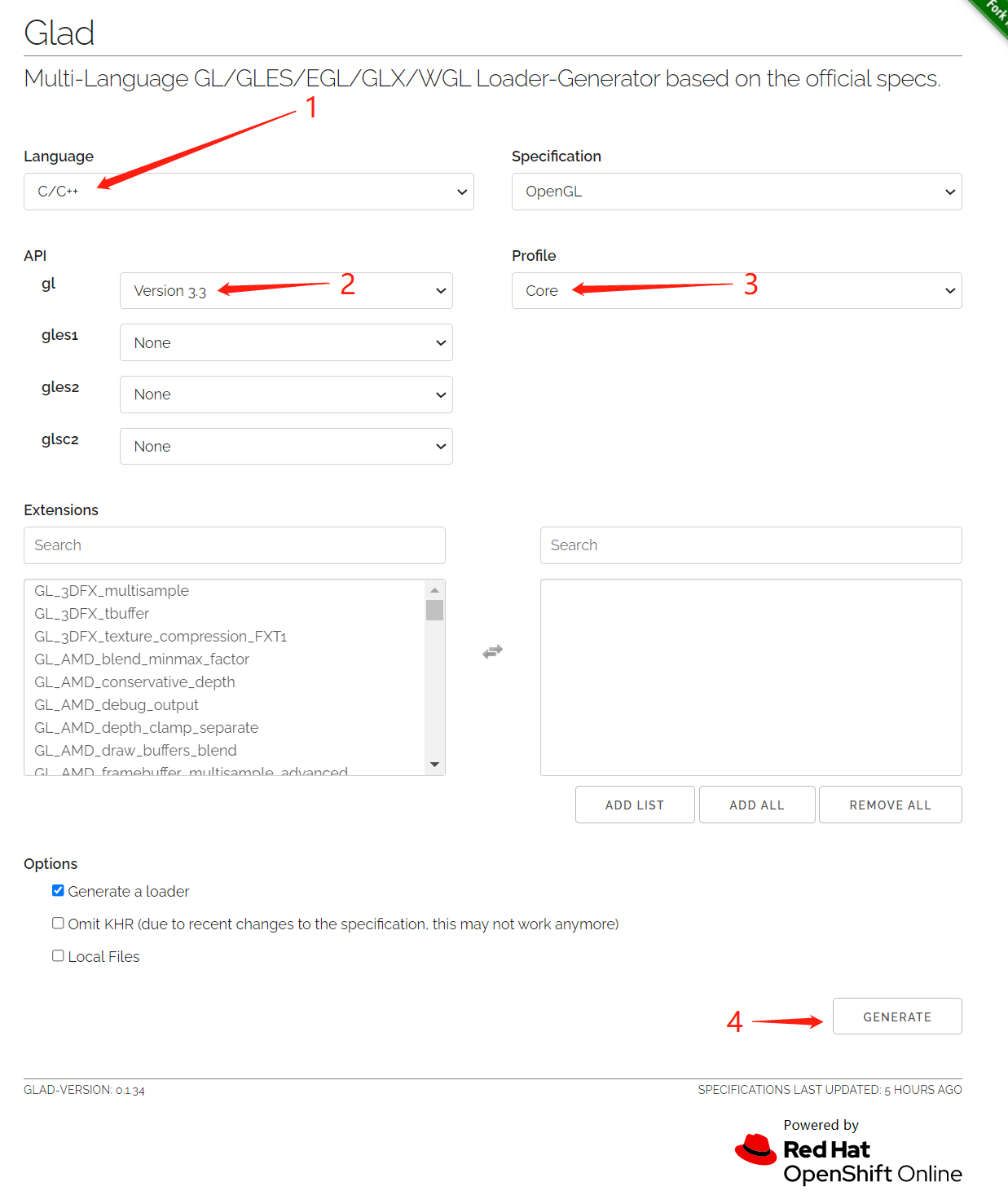
使用命令sudo apt-get build-dep glfw， sudo apt-get install cmake xorg-dev libglu1-mesa-dev安装编译GLFW所需要的依赖库。

在解压好的文件夹目录中使用 mkdir build命令创建一个新的文件夹存放编译文件，进入该文件夹后，使用 cmake ../ 以及 make命令编译源文件。如果没有报错，继续使用sudo make install 命令安装，出现以下信息代表安装成功。

****

（2）依赖库GLAD的安装

访问GLAD官网（<https://glad.dav1d.de/>）下载源文件。进去之后，选择语言为C/C++, API为Version 3.3，核心模式，之后点击生成。



下载生成的zip文件夹并解压，使用cd include 命令进入include文件夹，将子文件夹include中的内容移动到电脑 /usr/local/include文件夹中，对应的代码为sudo mv \* /usr/local/include。

（3）依赖库：GLM的安装

进入github（https://github.com/g-truc/glm）下载glm 的源代码（并解压，进入解压后的文件夹，可以发现一个glm文件夹，将此文件夹复制到/usr/local/include下即可，使用的命令为sudo cp -r glm/ /usr/local/include。

1. **使用g++编译运行代码**

新建项目文件夹，解压提供的参考代码，把内容拷贝到项目文件夹中。

（1）少量的代码修改

首先在Angel.h内添加以下代码。

#define fopen\_s(pFile,filename,mode) ((\*(pFile))=fopen((filename),(mode)))==NULL



在InitShader.cpp内添加#include <cstdlib>，并将 #include “Angel.h” 修改为#include “include/Angel.h”.



（2）在代码目录下，打开终端，输入命令：

g++ main.cpp InitShader.cpp glad.c -lglfw3 -lGL -lX11 -lpthread -lXrandr -lXinerama -lXcursor -lXi -ldl -o main

运行生成的可执行文件：./main

