

华东理工大学

计算机图形学

2023年9月

奉贤校区

图形软件的发展及软件标准的形成

- 图形软件的发展

开放式、高效率、标准化、集成化、智能化、学科交叉。

- 图形软件标准

与设备无关、与应用无关、具有较高性能。

- 近二十年中，国际标准化组织ISO已经批准和正在讨论的与计算机图形有关的标准有：

GKS、GKS-3D、PHIGS、CGM、CGI、IGES、STEP。

- 事实标准

SGI的OpenGL，微软的Direct X，Adobe的Postscript等。

OpenGL图形软件包

- OpenGL的主要功能
- OpenGL的绘制流程
- OpenGL的基本语法
- OpenGL的程序实例

OpenGL

- OpenGL (Open Graphics Library) 是一个功能强大的图形库，包括了120个图形函数，开发者可以用这些函数建立三维模型和进行三维实时交互。
- 用户可以很方便地利用它开发出有多种特殊视觉效果(如光照、纹理、透明、阴影)的三维图形。

OpenGL

- OpenGL是SGI (Silicon Graphics Inc.) 公司对IRIS GL进行改进, 扩展可移植性, 形成的一个跨平台开放式图形编程接口。
- 目前, OpenGL标准由1992年成立的独立财团OpenGL Architecture Review Board (ARB) 以投票方式产生, 并制成规范文档公布。
- **OpenGL绘制流水线**

Henry Ford's Big Idea:



现代流水生产起源于1914年—1920年的福特制

(一) 现代流水生产起源于1914年—1920年的福特制。

福特制的主要内容：

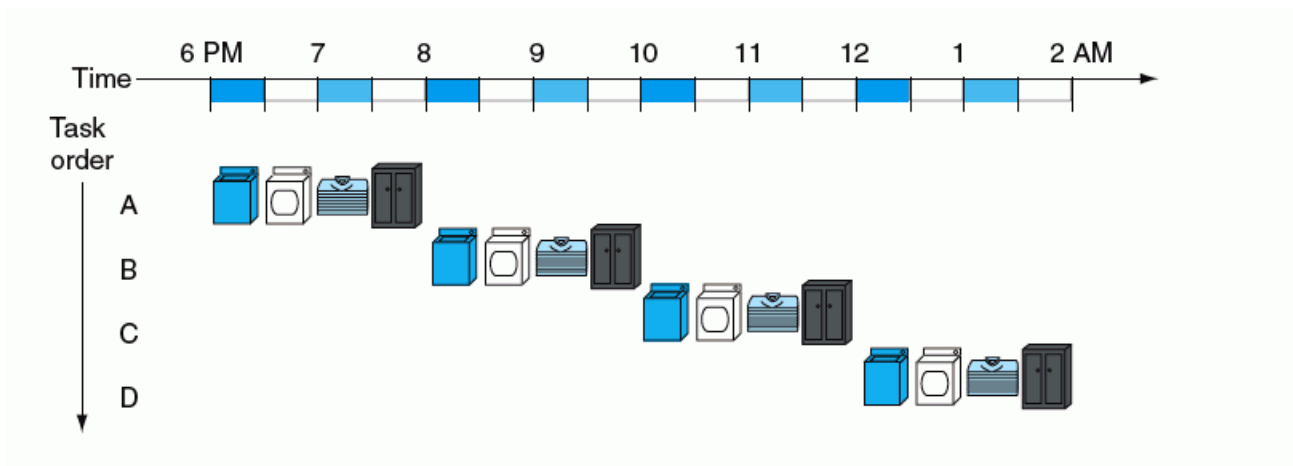
1.在科学组织生产的前提下谋求高效率和低成本。因而实施产品、零件的标准化、设备和工具的专用化以及工作场所的专业化。

唯一最佳的“单一产品原则”

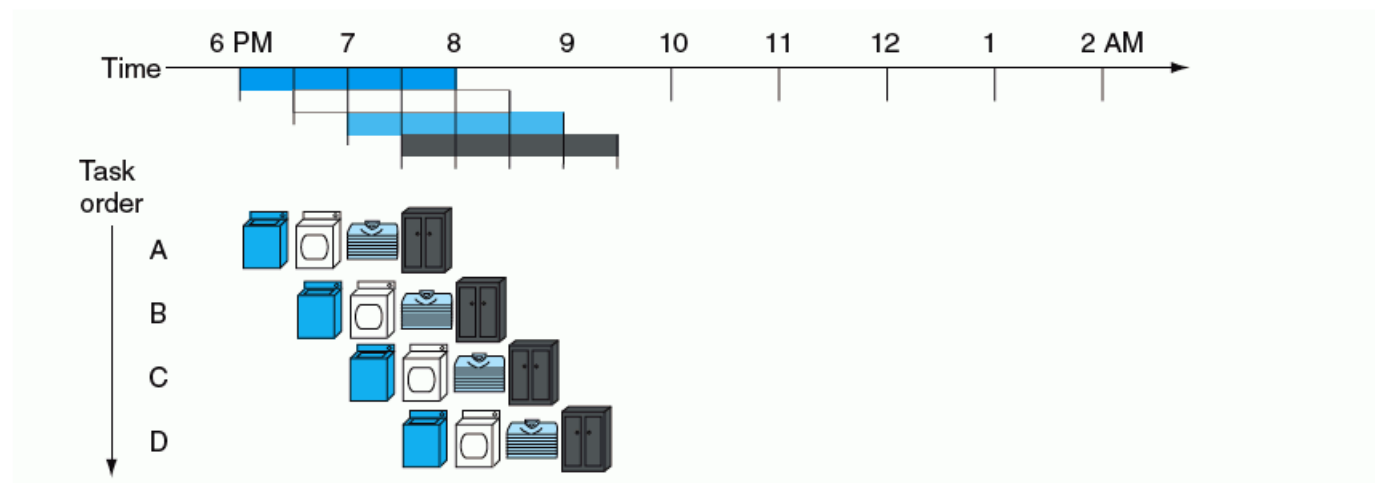
2.创造了流水线的生产方法，建立了传送带式的流水生产线。

流水线技术的基本原理

- 把一个重复的过程分解为若干子过程，每个子过程由专门的功能部件（模块）来实现。将多个处理过程时间上错开，依次通过各功能段，这样，每个子过程与其他子过程并行进行。

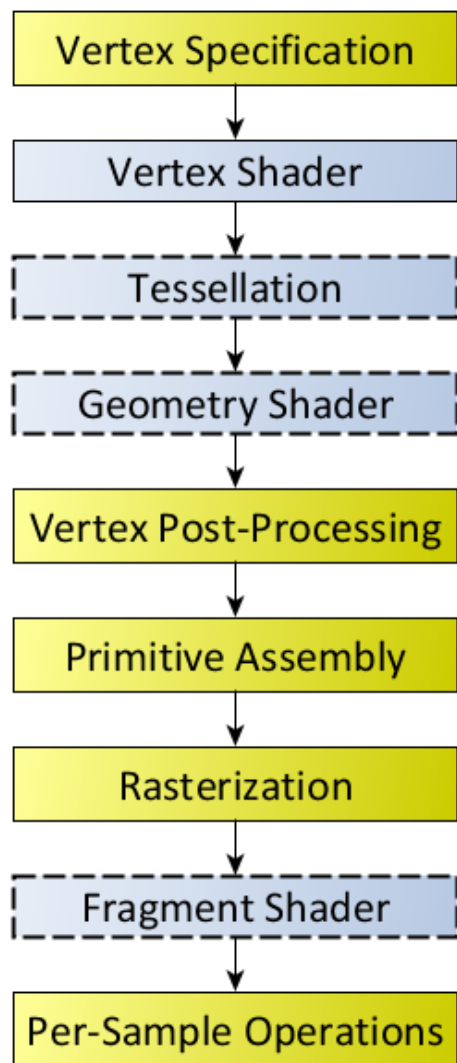
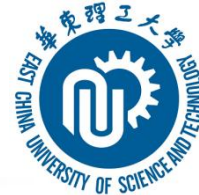


未用流水线的时间空间图



流水线时间空间图

OpenGL绘制流水线



1. Vertex Processing (**Vertex Shader**) :

- ① Each vertex retrieved from the vertex arrays (as defined by the VAO) is acted upon by a **Vertex Shader**.
- ② Optional primitive **tessellation stages**.
- ③ Optional **Geometry Shader** primitive processing

2. Vertex Post-Processing, the outputs of the last stage are adjusted or shipped to different locations.

- ① **Transform Feedback** happens here.
- ② **Primitive Assembly**
- ③ Primitive **Clipping**, the **perspective divide**, and the **viewport transform** to window space.

3. Scan conversion and primitive parameter interpolation, which generates a number of **Fragments**.

4. A **Fragment Shader** processes each fragment. Each fragment generates a number of outputs.

5. Per-Sample_Processing, including but not limited to: **Scissor Test / Stencil Test / Depth Test / Blending** **Logical Operation/ Write Mask**

OpenGL的主要功能

- 模型绘制
- 模型观察
- 颜色模式
- 光照应用
- 图像效果增强
- 位图和图像处理
- 纹理映射
- 实时动画
- 交互技术

OpenGL的绘制流程——工作方式

- 一个完整的窗口系统的OpenGL图形处理系统的结构
 - 最底层为图形硬件,
 - 第二层为操作系统,
 - 第三层为窗口系统,
 - 第四层为OpenGL,
 - 最上面的层为应用软件。

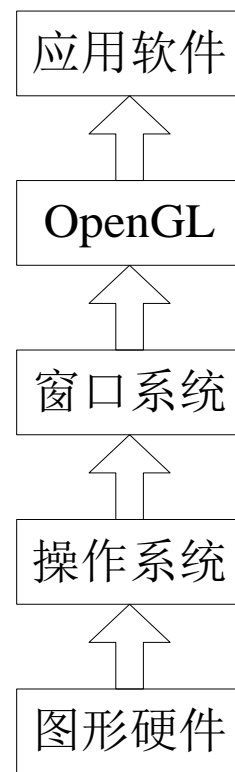
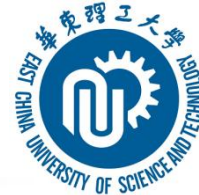


图2.34 OpenGL图形处理系统在计算机系统中的层次结构

OpenGL的绘制流程——流水线



- OpenGL命令将被放在一个命令缓冲区中，这样命令缓冲区中包含了大量的命令、顶点数据和纹理数据。当缓冲区被清空时，缓冲区中的命令和数据都将传递给流水线的下一个阶段。

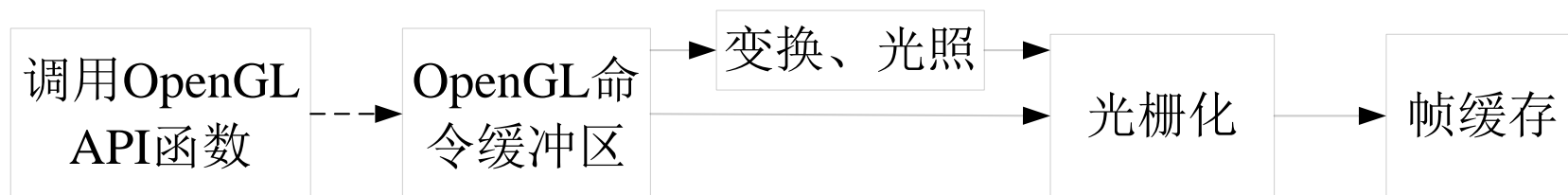
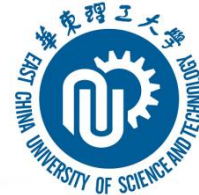


图2.35 一条简化的OpenGL绘制流水线

OpenGL的基本语法——相关库

- OpenGL核心库： gl
- OpenGL实用程序库： glu
- OpenGL编程辅助库： aux
- OpenGL实用程序工具包（OpenGL utility toolkit, GLUT）： glut
- Windows专用库： wgl

OpenGL的基本语法——命名规则



- OpenGL函数都遵循一个命名约定，即采用以下格式：

〈库前缀〉〈根命令〉〈可选的参数个数〉〈可选的参数类型〉

例如函数glColor3f(...), gl表示这个函数来自库gl.h, color表示该函数用于颜色设定, 3f表示这个函数采用了三个单精度浮点数参数。

OpenGL的基本语法——数据类型

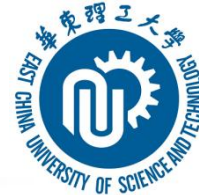


表2.1 OpenGL的数据结构

OpenGL数据类型	内部表示法	定义为C类型	C字面值后缀
GLbyte	8位整数	signed char	B
GLshort	16位整数	short	S
GLint, GLsizei	32位整数	long	L
GLfloat, GLclampf	32位浮点数	float	F
GLdouble, GLclampd	64位浮点数	double	D
GLubyte, GLboolean	8位无符号整数	unsigned char	Ub
GLushort	16位无符号整数	unsigned short	Us
GLuint, GLenum, GLbitfield	32位无符号整数	unsigned long	Ui

OpenGL程序实例——头文件包含

- 利用OpenGL实现图形绘制，首先要引入OpenGL核心库以及其他需要使用的库的头文件。
- GLUT保证了gl.h和glu.h被正确包含。

```
#include <gl/glut.h>
```

```
#include <windows.h>
```

OpenGL程序实例——窗口管理

- 初始化（`glutInit`）
- 创建窗口（`glutCreateWindow`）
- 设定窗口的显示模式（`glutInitDisplayMode`）
- 指定窗口的位置和大小（`glutInitWindowPosition`和`glutInitWindowSize`）
- 指定窗口的显示内容函数（`glutDisplayFunc`）
- 运行框架（`glutMainLoop`）

OpenGL程序实例——绘制图形

- 指定窗口背景色（ `glClearColor` ）
- 颜色管理，使用RGB颜色模型。一种颜色用红、绿、蓝三种颜色成分混合而成，每种颜色成分使用0.0到1.0之间的任意有效浮点数来表示颜色值。

OpenGL程序实例——绘制图形

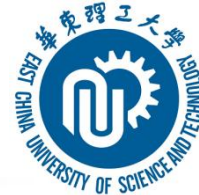


表2.2 OpenGL的一些常用混合色

混合色	红色成分 (R)	绿色成分 (G)	蓝色成分 (B)
黑	0.0	0.0	0.0
红	1.0	0.0	0.0
绿	0.0	1.0	0.0
黄	1.0	1.0	0.0
蓝	0.0	0.0	1.0
紫	1.0	0.0	1.0
青	0.0	1.0	1.0
深灰	0.25	0.25	0.25
浅灰	0.75	0.75	0.75
棕	0.60	0.40	0.12
南瓜橙	0.98	0.625	0.12
粉红	0.98	0.04	0.70
紫红	0.60	0.40	0.70
白	1.0	1.0	1.0

OpenGL程序实例——绘制图形

- 刷新窗口的缓冲区（ `glClear` ）

- 设定投影参数

```
glMatrixMode(GL_PROJECTION);
```

```
gluOrtho2D(0.0,200.0,0.0,150.0);
```

- 绘制图形

```
glRectf(50.0f, 100.0f, 150.0f, 50.0f);
```


OpenGL程序的交互功能

1. 可用GLUT实现;
2. 主要交互方式:
 - ① 键盘
 - ② 鼠标
 - ③ 菜单
 - ④ 拾取

OpenGL开发与程序语言无关

- C++
- Web (WebGL)
- Python
- Java
- ...