

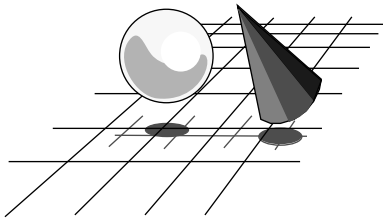
Programming with OpenGL: An Introduction

A TECHNICAL OVERVIEW OF THE OpenGL[®] GRAPHICS API

Course Notes for SIGGRAPH '96

Course Organizer

Tom McReynolds
Silicon Graphics Computer Systems



Course Speakers

Tom McReynolds
Silicon Graphics Computer Systems
Kathleen Danielson
Silicon Graphics Computer Systems

The text "OpenGL" is rendered in a colorful, 3D, blocky font. Each letter is a different color: 'O' is red, 'P' is green, 'e' is blue, 'n' is yellow, 'G' is purple, and 'L' is pink. The letters have a slight shadow and are arranged in a slightly staggered, perspective view.

Abstract

OpenGL, the standard software interface for graphics hardware, allows programmers to create interactive 2D and 3D graphics applications on a variety of systems. With OpenGL you can create high-quality color images. OpenGL makes it easy to build geometric models, change the viewing position, control the color and lighting of geometric primitives, and manipulate pixel and texture map images.

This course will cover an immediately applicable subset of OpenGL, so that you can write a simple graphics program, using shading, lighting, texturing and hidden surface removal. We will also discuss the new features in the most recent versions of OpenGL and GLX: OpenGL 1.1 and GLX 1.2.

About the Speakers

Tom McReynolds

Tom McReynolds is a software engineer in the Performer group at Silicon Graphics. Before that, he worked in the OpenGL group where he's implemented OpenGL extensions and done OpenGL performance work. Prior to SGI, he worked at Sun Microsystems, where he developed graphics hardware support software and graphics libraries, including XGL.

Tom is also an adjunct professor at Santa Clara University, where he teaches courses in computer graphics using the OpenGL library. Address: 2011 N. Shoreline Boulevard, Mountain View, CA 94043, E-mail: tomcat@asd.sgi.com, Phone: 415-933-5144, Fax: 415-965-2658

Kathleen Danielson

Kathleen Danielson is a software engineer in the OpenGL group at Silicon Graphics. At SGI she has implemented OpenGL extensions, improved OpenGL performance, and implemented Ada and Java bindings for OpenGL. Previous to SGI, she worked at Kubota Graphics in the graphics system software group, where she developed OpenGL libraries for several graphics hardware systems. Address: 2011 N. Shoreline Boulevard, Mountain View, CA 94043, E-mail: kat@asd.sgi.com, Phone: 415-933-1239, Fax: 415-965-2658

NOTICES

IRIS, Geometry Link, Geometry Partners, Geometry Accelerator, Geometry Engine, and OpenGL are registered trademarks of Silicon Graphics, Inc.

The X Window System is a registered trademark of the Massachusetts Institute of Technology.

DEC is a registered trademark of Digital Equipment Corporation.

IBM and OS/2 are registered trademarks of International Business Machines.

Motif is a trademark of Open Software Foundation, Inc.

Windows NT is a registered trademark of Microsoft, Inc.

UNIX is a registered trademark of AT&T Bell Laboratories.

The contents of this publication are subject to change without notice.

Table of Contents/Schedule

Introduction

Speakers	7
Objectives	7
What is OpenGL?	8
OpenGL State Machine	9
OpenGL Data Flow	10
OpenGL Main Points	11
OpenGL API Hierarchy	13
Utility Library (GLU)	14
OpenGL Extension to X (GLX)	15
More on GLX	16
GLUT (Graphics Library Utility Toolkit)	17
Structure of a Typical Program	18
OpenGL Command Syntax	19
States	20
Querying States	21
Drawing Geometry	22
Vertex Arrays	24
Display Lists	26
Code: Immediate Mode vs. Display List	27
Display List Editing	28
Culling	29
Colors	30
Processing of Colors	31
Point, Line, and Polygon Attributes	32
Transformations and Coordinate Systems	33
Transformation Flow	34
Transformations	35
Using Transformations	36
Before We Look at Some Code	37
An OpenGL Program	38

Break (elapsed time to here: roughly 1 1/2 hours)

Return from Break

Lighting and Shading 40
Lighting Properties 41
Lighting Code 42
Lighting Program 43
Hidden Surface Removal 46
Polygon Offset 47
Imaging 48
Drawing Bitmaps and Images 49
Pixel Operations 50
Histogram and Convolution Extensions 51
Color Table and Color Matrix Extensions 52
Blend Extensions 53
Imaging Pipeline with Extensions 54
Texture Mapping 55
Controlling Texturing 56
New Texture Functionality 58
Texturing Program 59
Atmospheric Effects 63
Antialiasing 64
Per Fragment Operations 65
Stencil Planes 67
Alpha Blending 68
Accumulation Buffer 69
Feedback & Selection 70
Evaluators/NURBS 70
OpenGL 1.1 features 71
GLX 1.1 and GLX 1.2 72
GLU 1.2 73
Extensions 74
Using Extensions and Versions 75
Summary 76
For More Information 77

End of Session (Question & Answers)