

PLATFORM INDEPENDENT GAME ENGINE

by

Chad Wesley Armstrong

A thesis submitted in partial fulfillment
of the requirements for the degree

of

Master of Science

in

Computer Science

MONTANA STATE UNIVERSITY
Bozeman, Montana

July 2003

APPROVAL

of a thesis submitted by

Chad Wesley Armstrong

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the College of Graduate Studies.

Ray Babcock

(Signature)

(Date)

Approved for the Department of Computer Science

Michael Oudshoorn

(Signature)

(Date)

Approved for the College of Graduate Studies

Bruce R. McLeod

(Signature)

(Date)

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Montana State University, I agree that the Library shall make it available to borrowers under rules of the Library.

If I have indicated my intention to copyright this thesis by including a copyright notice page, copying is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for permission for extended quotation from or reproduction of this thesis in whole or in parts may be granted only by the copyright holder.

Signature _____

Date _____

Dedicated to my
Great Aunt Betty Armstrong

VITA

Chad Wesley Armstrong was born in Longmont, Colorado on November 17, 1977. He is the firstborn to Donald and Dawn Armstrong. He grew up in Great Falls, Montana and lived there until 1996 when he moved to Bozeman, Montana to attend Montana State University. Four years later he received a Bachelor's degree in Computer Science and a Minor in German. He continued on to graduate school at Montana State University where he completed his Master's degree, also in Computer Science.

TABLE OF CONTENTS

| | |
|--------------------------------------|------|
| LIST OF TABLES..... | vii |
| LIST OF FIGURES..... | viii |
| ABSTRACT..... | ix |
| 1. INTRODUCTION..... | 1 |
| Related Work..... | 4 |
| 2. OPENGL..... | 6 |
| COLLISION DETECTION..... | 7 |
| Collision Detection Mathematics..... | 9 |
| Backface Culling..... | |
| Sphere - Plane Collision..... | 12 |
| PICKING AND SELECTION..... | 15 |
| LOADING TEXTURES..... | 18 |
| 3. OPENAL..... | 27 |
| 4. PORTING..... | 33 |
| 5. IMPLEMENTATION..... | 39 |
| 6. CONCLUSION..... | 43 |
| EVALUATION OF PIGE | 43 |
| FUTURE..... | 44 |
| BIBLIOGRAPHY..... | 46 |
| APPENDICES..... | 48 |
| APPENDIX A: PIGE Source Code..... | 49 |
| APPENDIX B: OpenAL Source Code..... | 87 |
| APPENDIX C: Texture Source Code..... | 94 |

LIST OF TABLES

| Table | Page |
|-------------------------------------|------|
| 1. Header examples..... | 2 |
| 2. Test computer systems..... | 3 |
| 3. alutLoadWAVFile differences..... | 35 |

LIST OF FIGURES

| Figure | Page |
|------------------------------------|------|
| 1. 2D grid..... | 8 |
| 2. Vector representation..... | 10 |
| 3. Plane normal..... | 1 |
| 4. Sphere - plane collision..... | 1 |
| 5. Cube without texture..... | 19 |
| 6. Cube with texture..... | 1 |
| 7. Texture mapping..... | 26 |
| 8. OpenAL demo screenshot..... | 28 |
| 9. OpenAL unhandled exception..... | 36 |
| 10. Psychomancer screenshot..... | 39 |
| 11. Game engine diagram..... | 4 |

ABSTRACT

The Platform Independent Game Engine (PIGE) has been designed as a set of tools to easily allow a developer to create a game which can be ported to multiple computing platforms with relative ease. PIGE makes use of cross-platform programming tools such as OpenGL, OpenAL, and the C language.

This gaming engine is separated into two main sections, the core code and the operating system specific header files. The core source code is platform neutral and requires no true modification when compiled and run. However, due to small inconsistencies between platforms, particular areas of PIGE require some functions to be written or modified for a particular operating system. This is where the OS specific header files come in, which are included and removed, depending on which system is running PIGE.