CROSS-PLATFORM OPEN SOURCE ISOMETRIC GAME DEVELOPMENT TARGETING THE ANDROID OPERATING SYSTEM

ANTHONY J. DIPERNA

THESIS FOR THE DEGREE OF MASTERS OF SCIENCE IN EMBEDDED SYSTEMS

OAKLAND UNIVERSITY

2011

Abstract

It is becoming increasingly difficult to stay competitive as a solo developer in the Mobile and Social gaming markets. Historically, one of the greatest aspects of software design is that an individual or small team of innovative developers could compete with much larger corporations. This window is, on one hand, closing as large corporations attempt to dominate the mobile market, and growing with the relatively recent rise of open source software. This thesis describes a game engine named IsoMob that enables developers to create isometric applications that target the World Wide Web, Desktop PC and Android simultaneously. It accomplishes this with a single code base written in HaXe. HaXe is an abstraction language that is able to compile to various languages such as C++, Actionscript 3.0, PHP, HTML5 and Javascript. The design and implementation of the game engine are described, as well as the results of an Application built using IsoMob.

Contents

1 Introduction

This thesis presents an isometric game engine that can target multiple platforms with a single code base. The engine is targeted at helping an individual or small team of developers quickly and efficiently build an application that runs on Google Android, Adobe Flash, and on any other C++ target. There have been recent advances in cross-platform mobile game development, but most of these solutions are targeted at either the very complex or simplest of games.

1.1 Motivation

The range of applications being deployed to mobile environments deployed today is considerably broader than it was even a few years ago. The huge growth in the mobile market was a combination of cheap hardware, abundant technology, and essentially the iPhone. Since the iPhone was released, Google introduced the Android operating system, HP acquired Palm (WebOS), Nokia partnered with Windows Phone, and Motorola Spun off Motorola Mobility Holdings [13]. This breakthrough defined a new type of software application almost overnight; the mobile application. At the same time, Facebook also became a viable development platform. Facebook transformed small development shops like Zynga, Playfish and Popcap games into game juggernauts that could compete with industry incumbents such as Electronic Arts (EA).

Small games,

- Mobile Development
  - Android
  - Growth Rate
  - Resources needed
  - Complexity
  - Speed To Market
  - Competition
  - One man shop
- Why Cross-Platform?

  - FaceBook
  - Kongregate
  - Problems it solves
  - Problems it creates
- HaXe
  - What it is
  - Why it was chosen
- Isometric
  - Filmation
  - Similar Applications
    * Bullet Hell

2 Related Work

2.1 Cross-Platform Strategy

2.1.1 Scripting

The Lua approach or Corona SDK / Moai

Using Lime, isometric support exists, but in beta state

2.1.2 Adobe Flash

2.1.3 Adobe AIR

2.1.4 HTML5

PhoneGap and Titanium

2.1.5 Consistent API

cocos2d, android uses Java, iPhone uses Objective-C, similar API

2.1.6 Limitations

- Corona SDK
  - Yearly Fee
  - Each additional piece of SDK costs more money
    - Lime
    - Spriteloq
  - No one has done anything isometric yet (20mil downloads)
- PhoneGap
  - HTML5 Based
- Titanium
  - HTML5 Based
- GameSalad
  - HTML5 Based

2.2 Native Application

C++ Application on Android

2.3 Development Environment

Testing software in FlashDevelop with Flash instead of using Emulator

2.4 Summary

Graph of competition and features

3 Engine Design

Rather than creating an entire game engine from scratch, I designed IsoMob based on proven game engine components. The Axonometric projection is based on the very first isometric game "Knight Lore" [17], AI is based on steering behaviors [16], the physics is a mix of Euler and Verlet Integration. This design decision facilitates the adoption of IsoMob with those familiar with game programming experience.

3.1 Isometry

3.1.1 Axonometric vs Dimetric

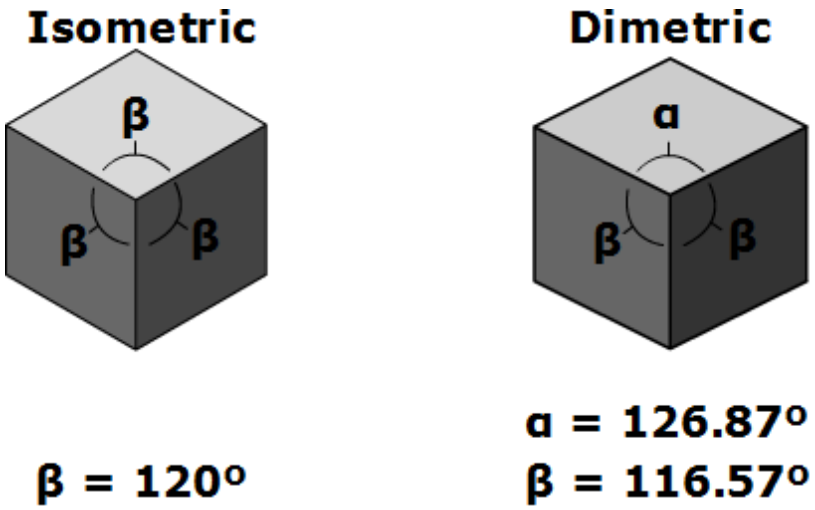

Figure 1: Isometric vs Dimetric Projection

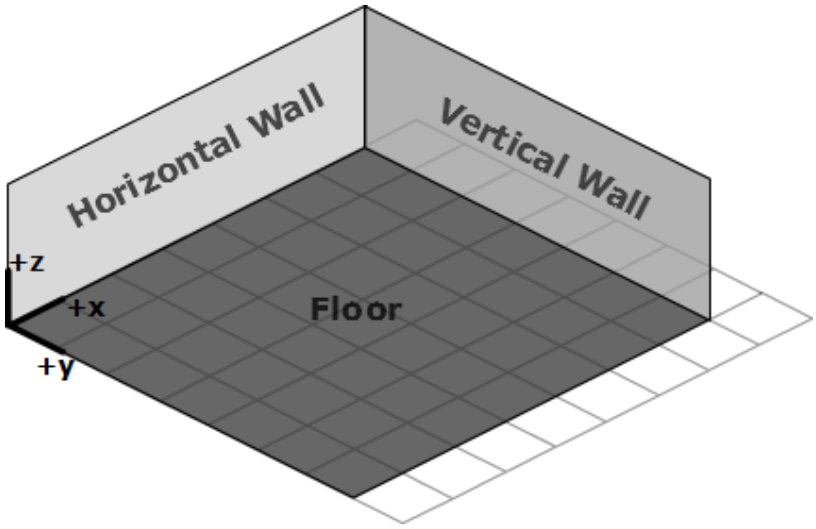

Figure 2: Isometric Projection

3.1.2 Isometric Transform

3.2 Scenes

3.2.1 Triggers

3.2.2 Extensible Markup Language (XML)

3.3 Walls

3.4 Floors

3.5 Entities

3.6 Resources

3.6.1 Platform Specifics

3.6.2 Tilesheets

3.6.3 Characters

3.6.4 Objects

3.6.5 Textures

3.7 Rendering

3.7.1 Platform Specifics

3.7.2 Depthsorting

3.7.3 RTrees

3.8 Saving State

[2] Peters, Keith. "AdvancED Actionscript 3.0 Animation". Apress (2008).

[3] Spuy, Rex. "AdvancED Game Design with Flash". Apress (2010).

[4] Dunn, Fletcher and Parberry, Ian. "3D Math Primer for Graphics and Game Development". Jones & Bartlett (2002).

[5] Cocos2d. "cocos 2d". http://cocos2d.org/

[6] Appcelerator. "Titanium". http://www.appcelerator.com/

[7] PhoneGap. "PhoneGap". http://www.phonegap.com/

[8] Ansca, Inc. "Corona SDK". http://www.anscamobile.com/corona/

[9] Twin-Motion. "HaXe". http://haxe.org/

[10] TonyPA. "Tile Based Games". http://www.tonypa.pri.ee/tbw/start.html

[11] OpenSteer. "Swarm Intelligence". http://opensteer.sourceforge.net/doc.html

[12] Zipline Games. "Moai". http://getmoai.com/

[13] GigaOM. "How Apple's iPhone changed everything". http://gigaom.com/2011/06/29/the-iphone-effect-how-apples-phone-changed-everything/

[14] CNN. "Why most people don't finish video games". http://articles.cnn.com/2011-08-17/tech/finishing.videogames.snow_1_red-dead-redemption-entertainment-software-association-avid-gamers?_s=PM:TECH

[15] Gatliff, Bill. "What is Google Android? An Introduction to Android Programming". ESC Chicago (2011).

[16] Reynolds, Craig. "Steering Behaviors for Autonomous Characters". Sony Computer Entertainment America (1999). http://www.red3d.com/cwr/steer/gdc99/

[17] Walker, Neil. "On Filmation". http://retrospec.sgn.net/users/nwalker/filmation/ (2004).