
GAME EXPERIENCE

Universe at War: Earth Assault (PC/360) – Petroglyph Games, Inc April 2007 - Current

Programmer

Worked on a cross platform engine doing PC and Xbox 360 tasks

- Worked on multiplayer matchmaking code and lobbies
- Worked with artists to implement GUI scenes using a LUA script based GUI system
- Ensured Games For Windows LIVE and Xbox 360 TCR compliance
- Created the installer for the PC SKU and created the Demo for the Xbox 360
- Implemented bug fixes in the engine and game
 - Became quite proficient debugging Power PC assembly code

Ghost Recon: Advanced Warfighter 2 (PSP) – High Voltage Software, Inc July 2006 - September 2006

Engine/Tools Programmer

Worked on a cross platform engine doing PC, PSP, and Xbox tasks

- Integrated a Flash UI middleware product on PC, PSP, and Xbox versions of the engine
 - Wrote PSP UI rendering code from scratch
 - Hooked the middleware product into the engine's memory management system
 - Integrated Flash into the tool chain so that it is automatically stripped and packaged for the engine
 - Worked with UI design team to ensure UIs fit within memory and needed features were properly supported
- Worked with artists to implement rendering bug fixes on the PSP and implement new functionality to conform with existing PS2 rendering code

Computer Assisted Storytelling in Video Games – Guildhall at SMU June 2006 – March 2007

Sole Programmer, Thesis Project

Utilizing AI techniques such as Goal Oriented Action Planning and pathfinding, developed a method for creating stories that change dynamically based on player interaction while still staying true to the designer's story vision

- Created a symbolic story representation based on the concept of locks and keys
- Formulated story elements as keys with preconditions and resultant outcomes
- Implemented an pathfinding search to satisfy the story goals as set forth by a designer

ndBot for Quake4 – Guildhall at SMU April 2006 – June 2006

Sole Programmer, Directed Focus Study

Added bot support for *Quake 4* multiplayer with awareness of jump-pads, teleporters, and desirable item pickups based on current bot state

- Implemented additions to the navigation system for teleporters and jump-pads
- Created a sensory perception system for items currently within the bots' view
- Developed a memory system for player and item locations containing a confidence metric

The Fiona Project – Guildhall at SMU October 2006 – March 2007

AI Programmer

A fifteen person team of artists, level designers, programmers, creating a highly stylized black and white 3rd person game project similar in style to *Gears of War* and *Devil May Cry* in the Source Engine

- Implemented coordinated squad attacks for enemies carrying shotguns, machineguns
- Developed behaviors for shotgun toting enemies that hound the player and move to cover in order to reload
- Developed behaviors for machine gun enemies that prefer taking pot shots from cover and will move from one cover spot to the next in order to flank enemies
- Produced a riot shield enemy that provides moving cover for shotgun and machine gun enemies

DangerBall '77 – Guildhall at SMU January 2006 – March 2006

Programmer/Team Lead

An eleven person team of artists, level designers, programmers, creating a 3rd person sports game project similar to roller derby for *Unreal Tournament 2004*

- Programmed many of the core subsystems such as skating and rail grinding

- Organized team to meet milestones set forth by the producer
- Directed artists, level designers, and programmers to produce a singular vision
- Coordinated between other team leads to setup gameplay testing schedule and bug feedback/tracking systems
- Worked on creation and updates of game documentation: Concept Document, Game Design Document, Technical Design Document, Level Design Documents, Art Style Guide, Asset and Development Plan

Lab Rats! – Guildhall at SMU

July 2005 – September 2005

Sole Programmer/Creator

A 2D game similar to the classic game *The Incredible Machine*, utilizing realistic physics in order to solve puzzles

- Implemented a 2D physics simulation using a particle constraint system which included rotational dynamics
- Created a graphics engine utilizing only Windows API and custom blitting (alpha blending, parallax scrolling)
- Optimized sprite rotations using indexed pre-rotated sprites because hardware acceleration was not desirable

Interpreted Scripting Language – Guildhall at SMU

April 2006 – May 2006

Sole Programmer

A state machine based weakly-typed scripting language with string and vector primitives

- Wrote a grammar and lexical analyzer for the language
- Implemented a parser to create a syntax tree from the token stream
- Built a compiler to output bytecode from the syntax tree
- Created a virtual machine to run the language

Ludibrium Game Engine – Guildhall at SMU

October 2005 – June 2006

Sole Programmer/Creator

3D game engine utilizing DirectX9 and OpenGL

- Utilizes a uniform way of rendering regardless of underlying API
- Loads BSP: lightmaps, diffuse texture, ray casting, PVS
- Renders large scale terrains using a patch based lod approach
- In-engine terrain editor: variable sized brushes, terrain splatting, quadtree subdivision
- Features: ARB shader language, HLSL, Particle Manager, 3DS Loader, LDM (custom mesh format), skeletal animation
- Implemented and tested a math library which includes vectors, quaternions, matrices, bounding volumes, and intersection tests

Compiler – Purdue University

August 2002 – December 2002

Sole Programmer

A compiler written in Java which outputs MIPS assembly for a language combining features from both Pascal and C

- Created a lexical analyzer, abstract syntax tree, and parser for a given grammar
- Developed an intermediate representation tree with associated interpreter
- Wrote optimizer utilizing peephole optimizations, common subexpression elimination, and smart register allocation
- Built a compiler that output MIPS assembly

EDUCATION

Graduate Certificate of Interactive Technology in Digital Game Development

March 2007

Guildhall at Southern Methodist University

Bachelor of Science in Computer Science and Mathematics

December 2004

Purdue University

SKILLS

Languages – C/C++, Java, Scheme, LUA, Power PC assembly, MIPS assembly, ARB Shader Language, HLSL, Unreal Script, PHP, HTML

Software – Visual Studio .NET 2003, SN Systems, Subversion, Alienbrain, Perforce, CVS, Adobe Photoshop CS2, Microsoft Office .NET 2003, 3D Studio Max 8, Microsoft Project 2003, vi

Platforms – PC, 360, PSP, Xbox, PS2, DirectX, OpenGL, Microsoft Windows, Solaris, Linux, FreeBSD, Cygwin

Physics – Newtonian dynamics, Runge-Kutta, Implicit/Explicit Euler, Collision detection, Collision response

Artificial Intelligence – A*, Planning, FSM, Game Tree Search, Constraint Satisfaction, Propositional Logic