Quake III : Fun And More

A tutorial introduction to building Quake III mods and more!

Shree Kumar

Hewlett Packard

Audience Poll

- How many of you
  - Have played a 3D game
  - Have done some 3D coding
  - Are artists
  - Have done some artwork for 3D games
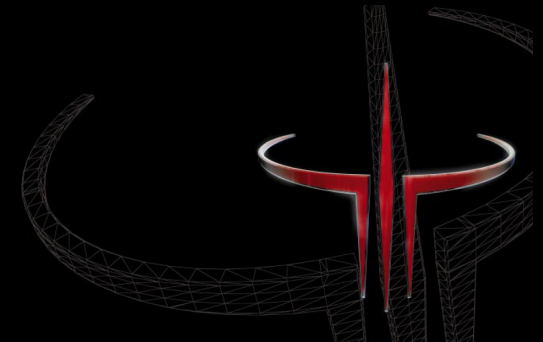 - Have made a mod ?
  - Still remember some math

Why Quake III

- Top-notch commercial engine
- Recently open-sourced
- Large community
- Lots of tools available
- Great opportunity to learn…

Tutorial Methodology

- Learn while having fun

- Q3 overview

- Simple examples
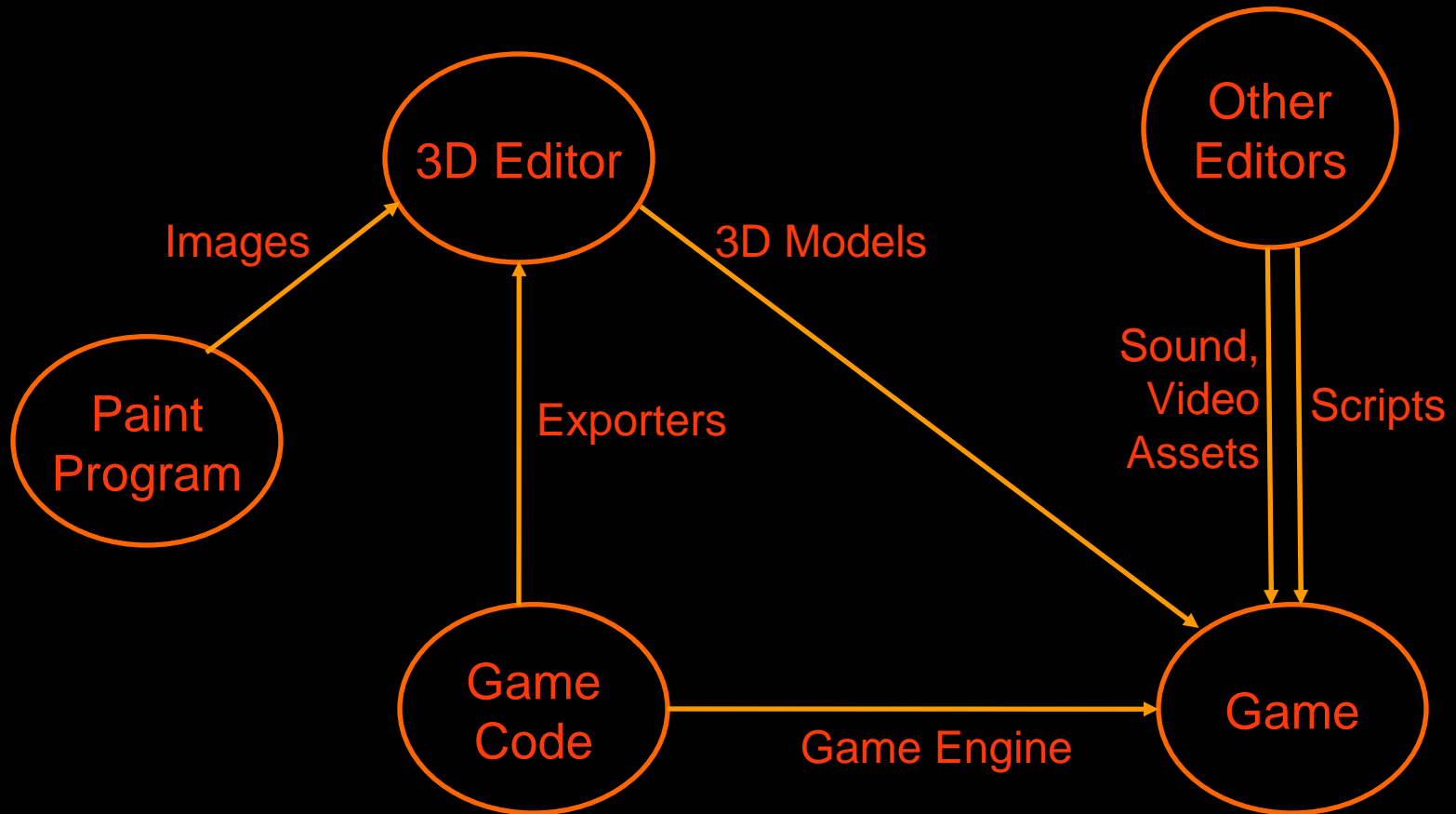 - Involved examples work with hands-on-sessions

Tutorial Outline

- General intro

- Your own level inside Q3

- Code changes 1-2-3

- Diving deeper

- Getting started on your own

- Conclusion

Game Making

3D Editor

Other Editors

Images

3D Models

Paint Program

Exporters

Sound, Video Assets

Scripts

Game Code

Game

Game Engine

The Quake III Engine

- Quake-C, VMs
- Principal Components
  - game
  - cgame
  - ui
  - renderer
- Features

Quake "Filesystem"

- .pk3 files
  - ZIP file
- Directory conventions
  - maps : .bsp, .aas
  - vm : .qvm files
  - env, gfx, icons, sounds, sprites, textures, models : .md3, .skin
  - scripts : .shader, arenas.txt, bots.txt
  - videos, demos

Tools Of The Trade

- Level editing : Radiant

- Bots : BotEdit

- Bot models : Milkshape3D, 3DS MAX (not free   )

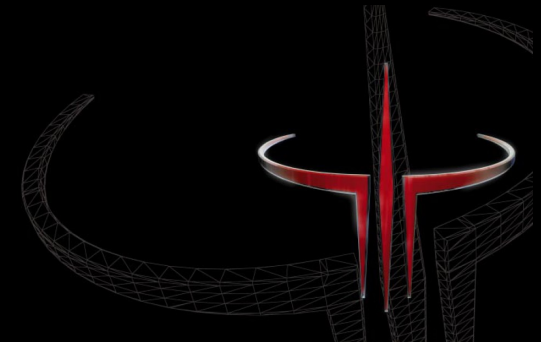- Image editing software, e.g. GIMP

- A text editor !

Level Editing – The Basics

- Entities

- You draw "brushes", not polygons

- Make sure your textures are included

- Size does matter

- Radiant isn't exactly your typical editor
  - Read the manual
  - Using other editors

Level Editing – The Essentials

- Modelling
  - Avoid "leaks"
  - info_player_start
- Compiling the map
  - Compilation options
  - Speed versus quality

Getting The Map Inside

- Creating the .aas file
- "packing" the media
- Viewing the level
    - ./quake3 +map testlevel
- arenas.txt

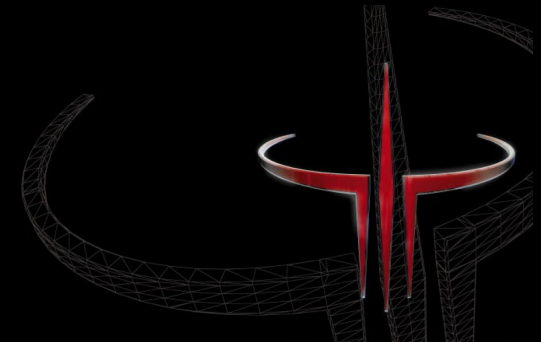# Source Layout

- code
  - unix, win32, macosx : platform support
  - cgame, game, q3_ui : VMs
  - client, server
  - qcommon, splines
  - renderer
- utilities

The Game Loop

```
┌──────────────────────────┐   Player State Updates    ┌──────────────────────────┐
│ cgame vm                 │ ─────────────────────────▶ │ game vm                  │
│  ┌────────────────────┐  │                            │  ┌────────────────────┐  │
│  │ game client        │  │ ◀───────────────────────── │  │ server, game logic │  │
│  └────────────────────┘  │  World State Update, Events│  └────────────────────┘  │
└──────────────────────────┘                            └──────────────────────────┘
         │
         │ Updates
         ▼
┌──────────────────────────┐
│ renderer                 │
└──────────────────────────┘
```

Basic Math

- Points

- Vectors
  - Normalization

- Matrices

- Quaternions

Your First Mod

- Hello Mod : Rocket Slowdown
  game/g_missile.c : fire_rocket()

```
VectorScale(dir, 900, bolt->s.pos.trDelta);
```
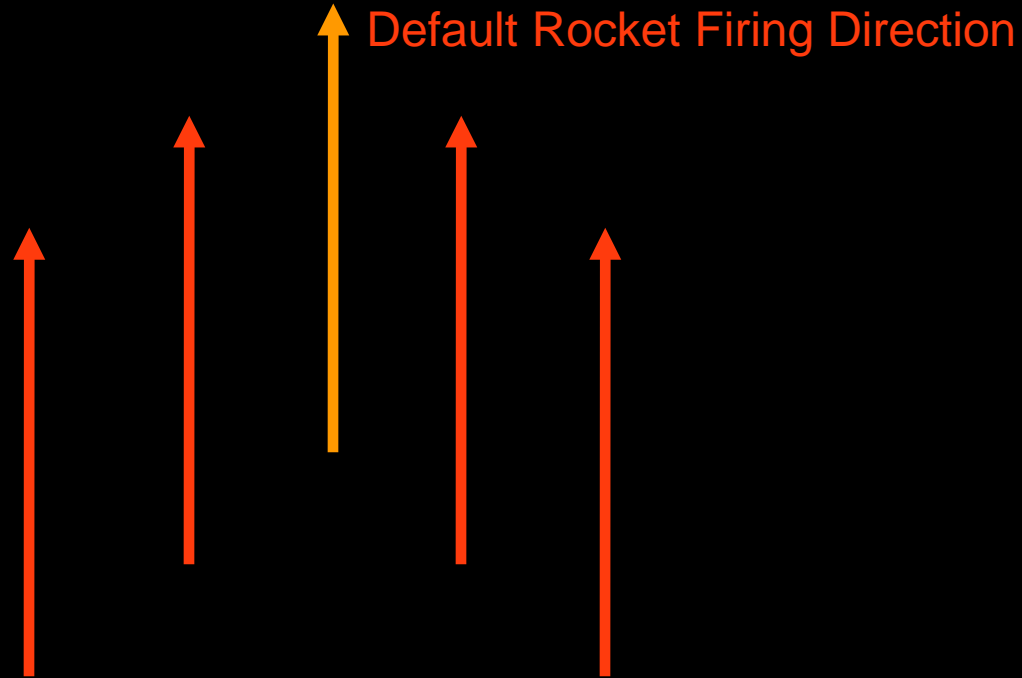
↓

```
VectorScale(dir, 100, bolt->s.pos.trDelta);
```

- Getting the code inside
  – Compile and copy
  – Beware .pk3 ordering
  – Debugging VMed code

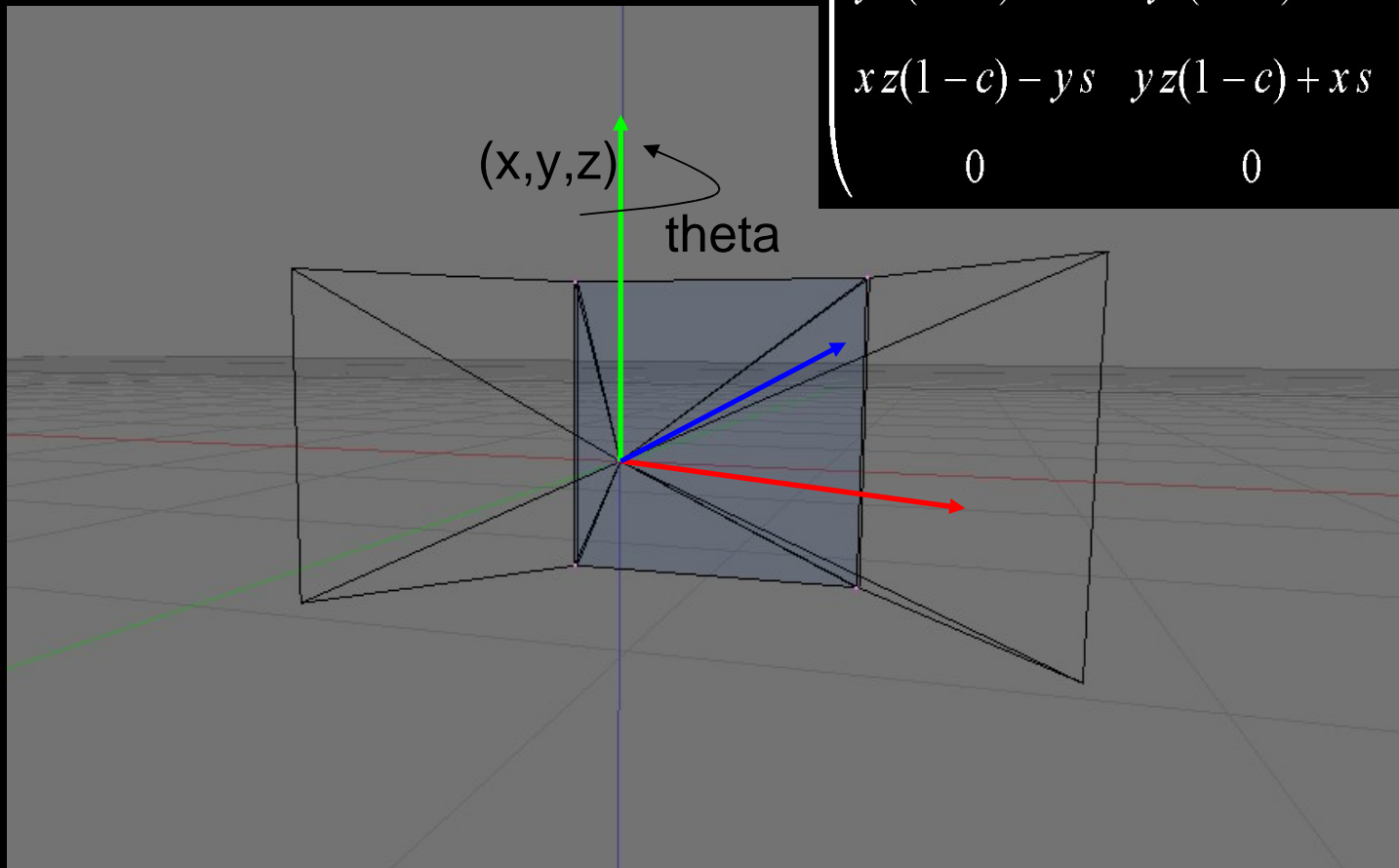# Quake-ayan

Default Rocket Firing Direction

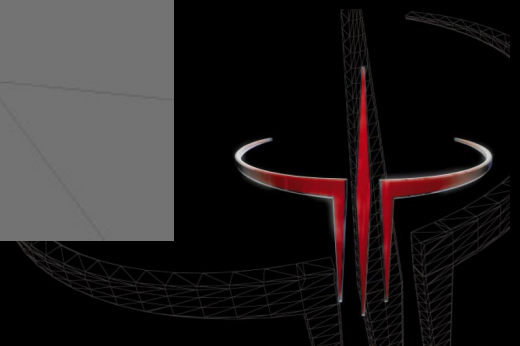game/g_weapon.c : Weapon_RocketLauncher_Fire()

View Alteration

$$\begin{pmatrix} x^2(1-c)+c & xy(1-c)-zs & xz(1-c)+ys & 0 \\ yx(1-c)+zs & y^2(1-c)+c & yz(1-c)-xs & 0 \\ xz(1-c)-ys & yz(1-c)+xs & z^2(1-c)+c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

(x,y,z)

theta

s=sin(theta)
c=cos(theta)

View Alteration (contd)

- Changes cgame
- Works in "true" networked mode
  - Ensure same VMs on all nodes or +sv_pure 0
- Game setup for CAVEishness
  - Connect to control server
  - follow player at center
- Mirror Detour

Doing an "Open Quartz"

- Source is free, not art

- To have a truly free engine, we need base art assets

- What it takes to create assets
  - Textures
  - Maps, Bots, other MD3s
  - Sounds
  - Game Logic

Not Just For Shooters

- Flexible Engine
    - Even has terrain support
    - Drastic mods : Quake Rally
- Learn from others
- One step at a time…

Getting Started

- Art
  - "Programmer Art"
  - Getting an artist
    - Low polygon pitfalls

- Science
  - Gameplay is important
  - Effects are cool (see tenebrae)

Links

- Source Code : www.idsoftware.com

- www.planetquake.com

  – Polycount, mod tutorials

- Tools
  – Radiant : www.qeradiant.com

  – Blender ?

    - So near, yet so far

  – Milkshape 3D, 3D Studio Max (not free)

- Google

Credits

- Id Software, for Quake
- Quake community

Questions ?

Shree Kumar

shreekumar3d@yahoo.com

(and soon http://www.shreekumar.in/)

Backup Slides

"Dodge-Ball"

- The rules
  - We play with greandes, not balls
  - Score before first bounce only
  - After first bounce, the ball passes on…
  - Slow weapon respawn
  - Identifying the hunters