Connecting Visuals to Gameplay at Valve

Jason Mitchell
Introduction

• Team Fortress 2
  • Distinctive Silhouettes
  • Stylized shading
• Left 4 Dead
  • Creating a Dark, Gritty Horror experience
  • Applying lessons learned from TF2
  • Utilizing “Filmic” effects
Team Fortress Mod
Initial Team Fortress 2
Team Fortress 2
Why The Unique Visual Style?

- Gameplay
- Readability
- Branding
Read Hierarchy

- **Team** - *Friend or Foe?*
  - Color
- **Class** - *Run or Attack?*
  - Distinctive silhouettes
  - Body proportions
  - Weapons
  - Shoes, hats and clothing folds
- **Selected weapon** - *What’s he packin'?*
  - Highest contrast at chest level, where weapon is held
  - Gradient from dark feet to light chest

Color Swatch
Rim Highlighting
Rim Highlighting
Character Creation

1. Character silhouette
2. Interior shapes
3. Model sheet
4. 3D Model
5. Character Skin
6. Final Character in game
Character Silhouette

- Building block of character design
- Identifiable at first read
Interior Shapes

- Solving interior character design with shadow shapes
- Keep it iconic
- Work out design in three quarter pose
Model Sheet

- Use concept painting as guide
- Solve design problems using silhouette only
- Solve interior design with shadow shapes
3D Model

- Match silhouette to model sheet
- Solve 3 quarter design with screenshots / paintovers
- Model with character in mind
Base Ambient Occlusion map
Character Skin
Final Character

- 3D model with texture and basic shading
Engineer Concept
Pyro Concept
Pyro model
Environment Design

- Creating a compelling, immersive world
- Team distinction through material hue/value/saturation
  - Desaturated relative to players
- Impressionistic painterly look
Contrasting Team Properties

- Red
  - Warm colors
  - Natural materials
  - Angular geometry

- Blue
  - Cool colors
  - Industrial materials
  - Orthogonal forms

Concept paintings
Miyazaki - Brush Width Foreshortened

- Can easily imagine a 3D camera move between these 2D views of the same space

Background plates from Spirited Away
World texturing

Texture map

In-game Screenshot
World texturing

Texture map

In-game Screenshot
World texturing

Texture map

In-game Screenshot
Introduction

- Co-op, first-person horror game
- Dynamic shared narrative
  - Experience an action movie with friends
- AI Director
  - Procedurally generated character performance, pacing, effects and music
- Shipped today!
  - (Please stay in your seats)
We Shipped Today!

- Because of Steam, there is an exact moment when the PC version of a Valve game officially ships.
- Hitting enter on a keyboard in one guy’s office wasn’t momentous enough for us, so we built The Shipping Machine
- I was here in Montreal, so I missed the party, but here are a few fun photos...
The Valve “Shipping Machine”
The Valve “Shipping Machine”
Left 4 Dead goes live!
Art Direction and Gameplay

- Dark, scary cinematic environment
- Apply lessons learned from TF2
- “Filmic” Effects
- Shaders enhance dark setting
Filmic effects

- Color Correction
- Grain
- Vignette
- Local Contrast Enhancement
- Dynamically communicate game state
No Post-processing
Color Correction
Before Vignette
Bathers at Asnières by George Seurat
Filmic Effects OFF
Filmic Effects ON
Normal Stress
High Stress
Hunter Pounce
Normal State
Third Strike
Lighting for Darkness

- Support fiction
  - Fires
  - Headlights of abandoned vehicles
- Aid navigation
  - Players tend to follow the light
- Importance of silhouette
- Player as light source
  - Flashlight tied to gameplay
Too many areas of contrast

Early production screenshot
Simplified lighting

Final look
In-game headlights
In-game headlights
Smoking the Set

- Separate foreground from background
  - Fog
    - Light colored fog in dark areas to contrast with silhouettes of infected in mid-ground
  - Particles
    - Adds atmosphere and helps accentuate silhouettes of infected against lighter particles
Black Fog

Light Fog
Reload, Shove & Muzzle Flash

- Player is the light source
- Increases drama and immersion
- Flashlight is attached to the weapons
  - Reloading
  - Shoving
  - Muzzle flash
- Encourages players to coordinate actions
Traditional Normal Mapping

• Traditional normal mapping locally alters surface orientation, causing detailed lighting effects
Self Shadowed Normal Mapping

- Self Shadowed Normal Mapping incorporates local self-shadowing information for greater surface richness
- Reacts to lighting from radiosity as well as dynamic lights in the scene, such as the player’s flashlight
- Refactoring our shader code, this turns out to be free
Self Shadowed Normal Mapping Example

No Self-shadowing

With Self-shadowing
Wet Environments

- Film technique
  - Wash down the set to get that “movie dark” look
- Film Noir
- Adds details to dark settings while still feeling dark
In-game screenshot
Summary

- Team Fortress 2
  - Distinctive Silhouettes
  - Stylized shading
- Left 4 Dead
  - Creating a Dark, Gritty Horror experience
  - Applying lessons learned from TF2
  - Utilizing “Filmic” effects