

Stylization with a Purpose

The Illustrative World of Team Fortress 2

Jason Mitchell





TEAM FORTRESS 2

OUTLINE

- History of Team Fortress
- Characters
 - Art direction
 - Shading algorithms
- Environments
- Meet the Team
- Post-ship
- Sneak Peek of the next *Meet the Team* short!



TEAM FORTRESS 2

TEAM FORTRESS MOD





TEAM FORTRESS 2

INITIAL TEAM FORTRESS 2





INITIAL TEAM FORTRESS 2





TEAM FORTRESS 2





TEAM FORTRESS 2

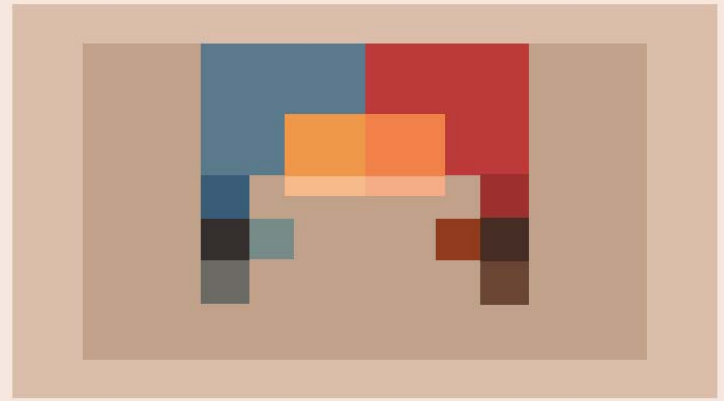
WHY THE UNIQUE VISUAL STYLE?

- 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



EARLY 20TH CENTURY COMMERCIAL ILLUSTRATION



Dean Cornwell



J. C. Leyendecker



Norman Rockwell



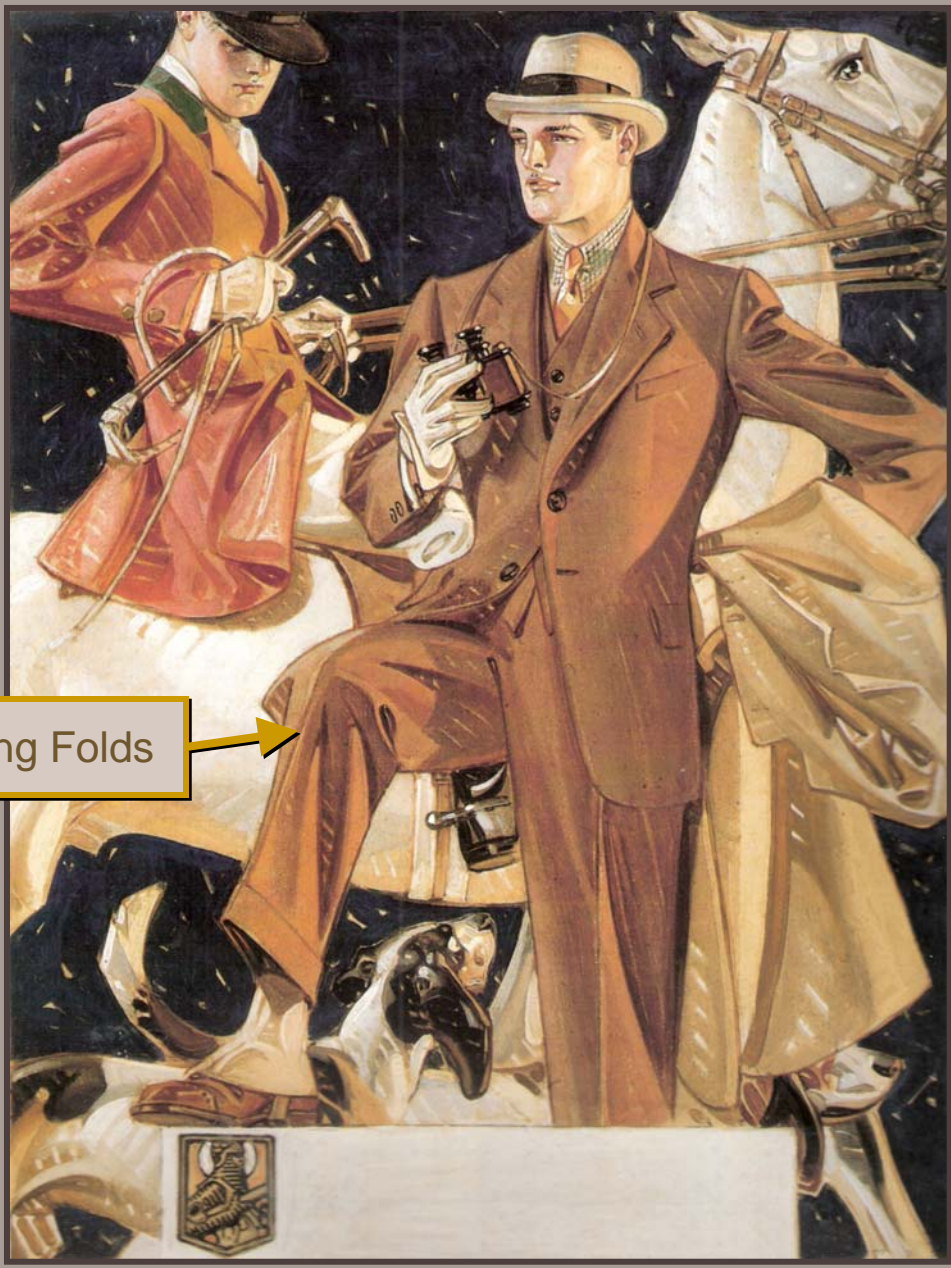
TEAM FORTRESS 2

EARLY 20TH CENTURY COMMERCIAL ILLUSTRATION

- Chose to adopt specific conventions of the commercial illustrator J. C. Leyendecker:
 - Shading obeys a warm-to-cool hue shift. Shadows go to cool, not black
 - Saturation increases at the terminator with respect to a given light source. The terminator is often reddened.
 - On characters, interior details such as clothing folds are chosen to echo silhouette shapes
 - Silhouettes are often emphasized with rim highlights rather than dark outlines



J.C. Leyendecker
Thanksgiving 1628-1928



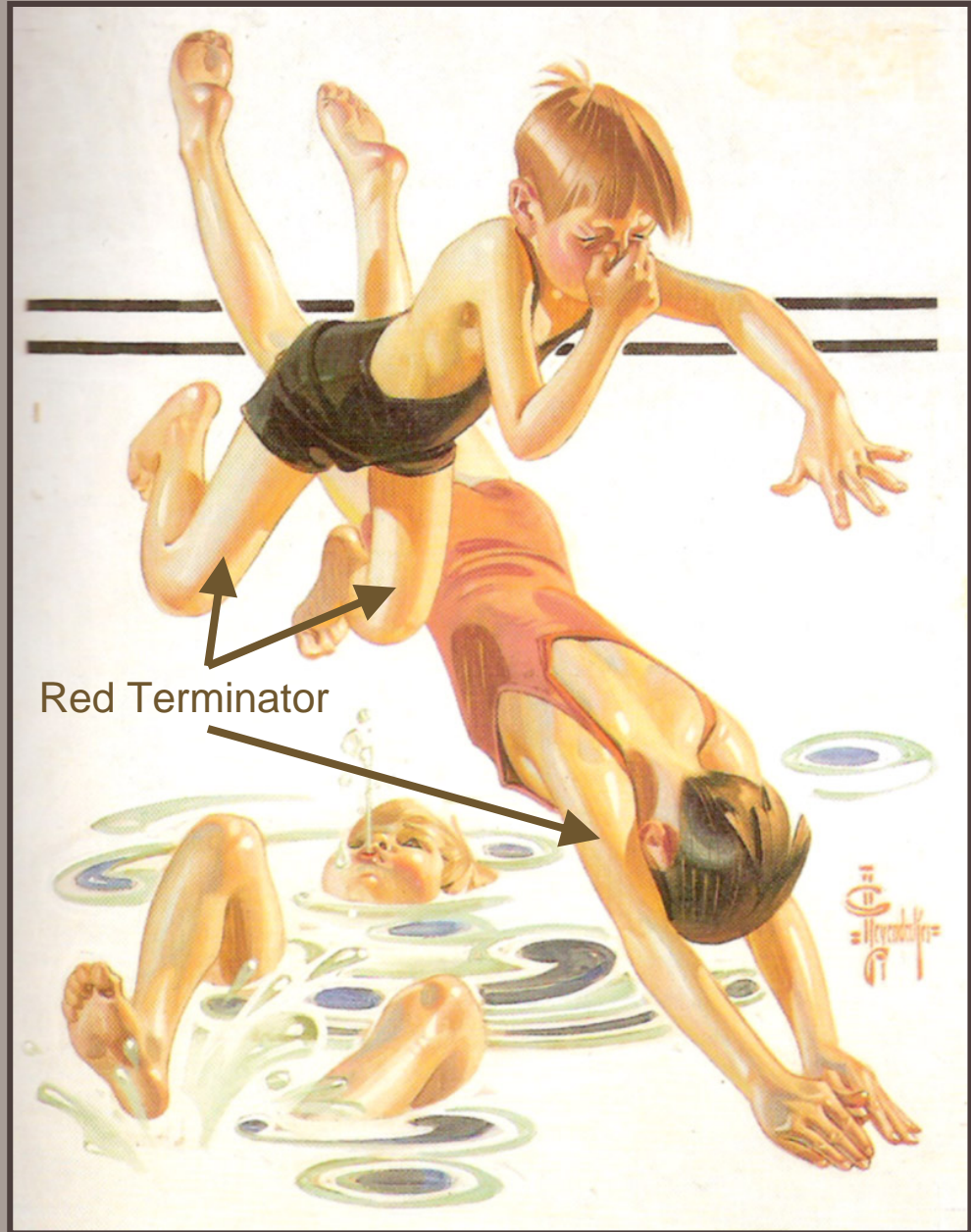
J.C. Leyendecker
Tally-Ho, 1930



Rim Highlights

J.C. Leyendecker

Arrow collar advertisement, 1929



Red Terminator

J.C. Leyendecker

Swimmin' Hole, 1935



RIM HIGHLIGHTING





RIM HIGHLIGHTING





TEAM FORTRESS 2

CHARACTER CREATION

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





TEAM FORTRESS 2

CHARACTER SILHOUETTE

- Building block of character design
- Identifiable at first read





TEAM FORTRESS 2

INTERIOR SHAPES

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

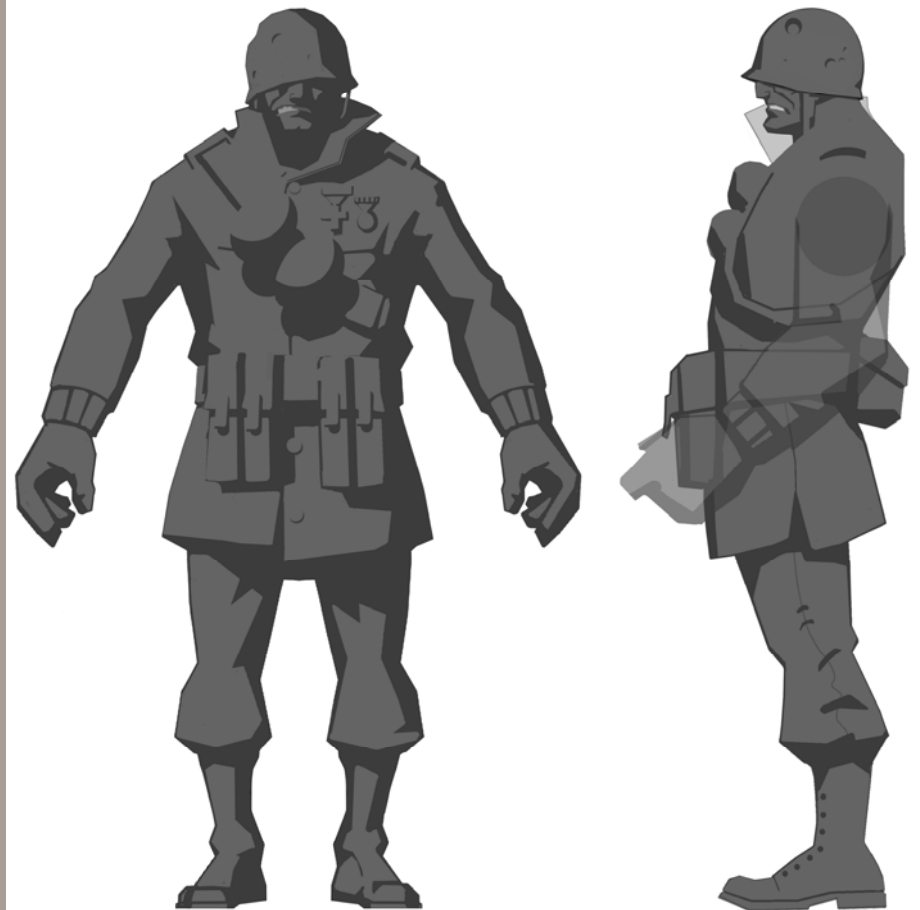




TEAM FORTRESS 2

MODEL SHEET

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

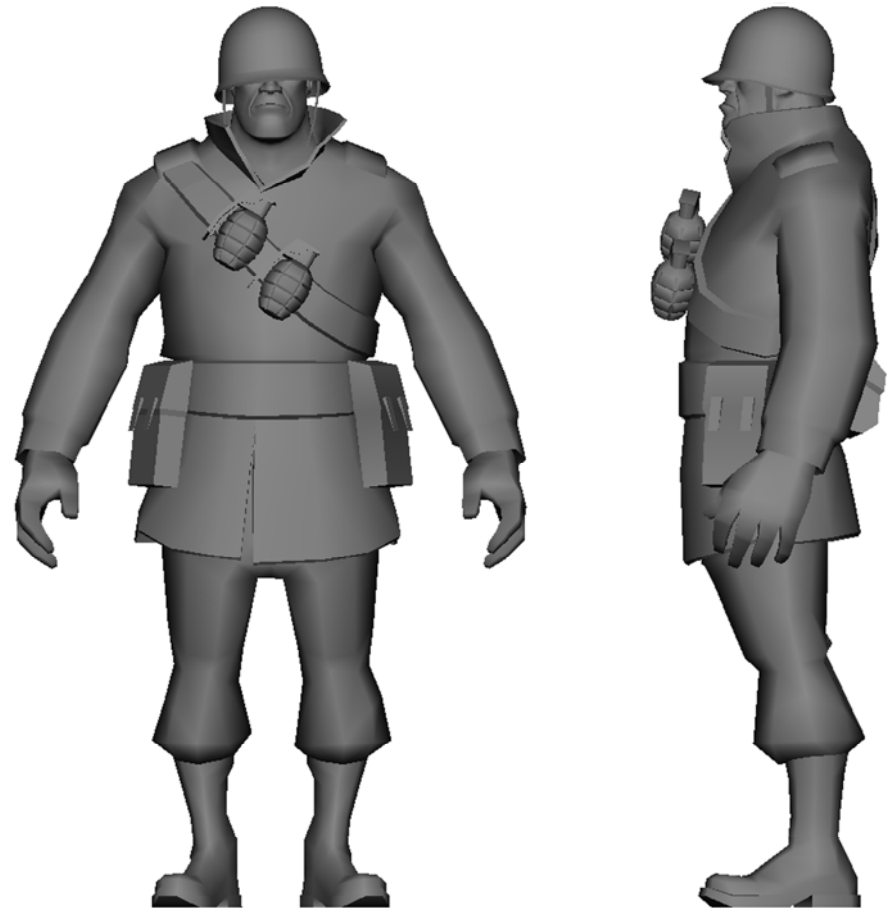




TEAM FORTRESS 2

3D MODEL

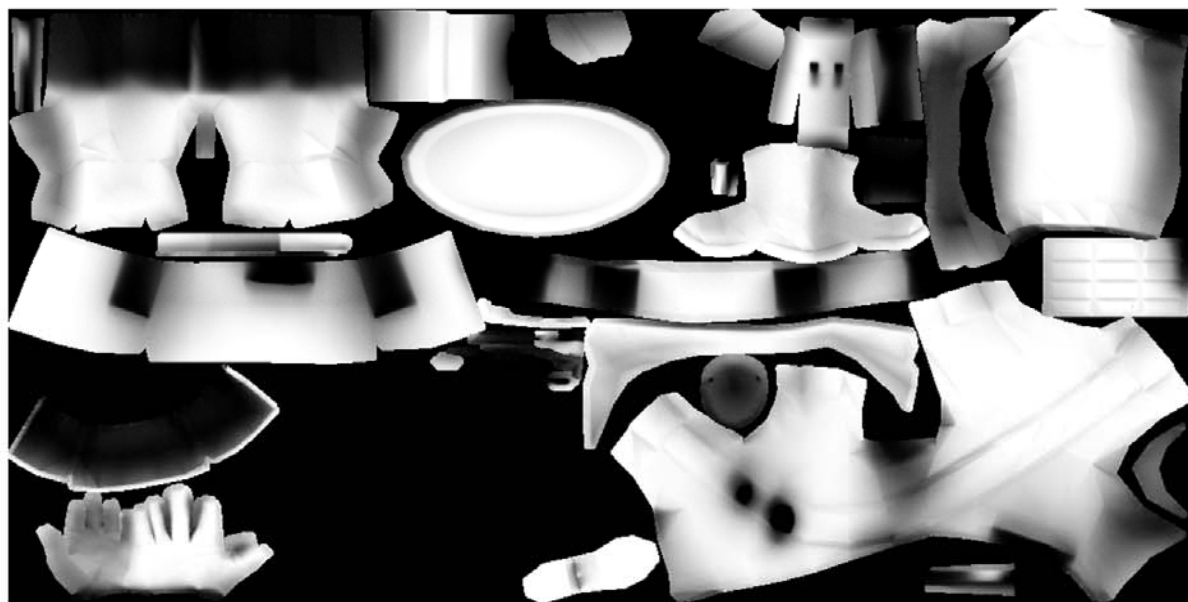
- Match silhouette to model sheet
- Solve 3 quarter design with screenshots / paintovers
- Model with character in mind





TEAM FORTRESS 2

BASE AMBIENT OCCLUSION MAP





CHARACTER SKIN





TEAM FORTRESS 2

FINAL CHARACTER

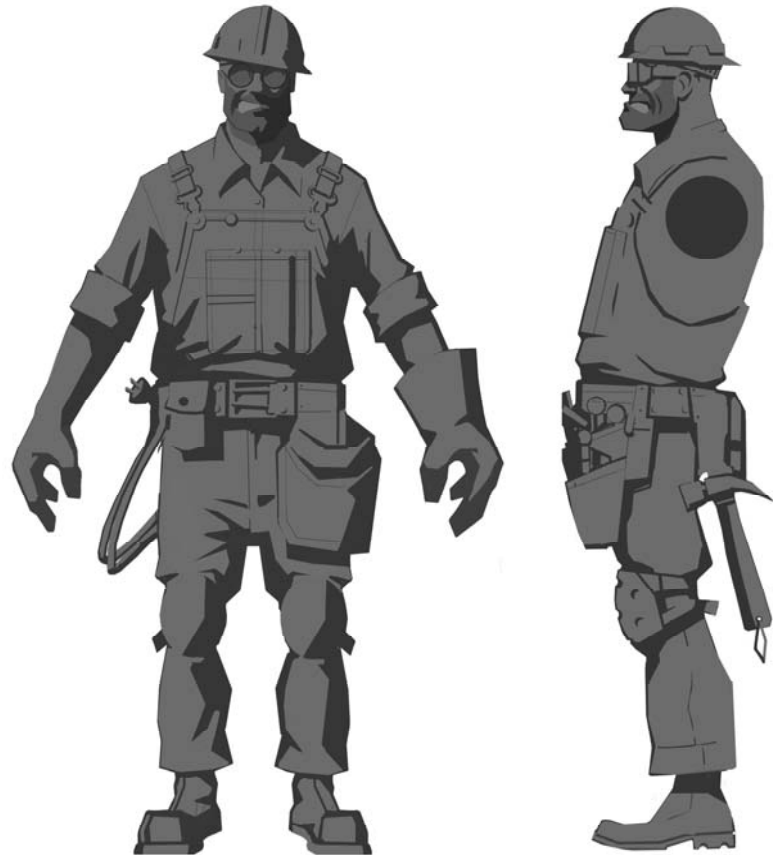
- 3D model with texture and basic shading





TEAM FORTRESS 2

ENGINEER CONCEPT





TEAM FORTRESS 2

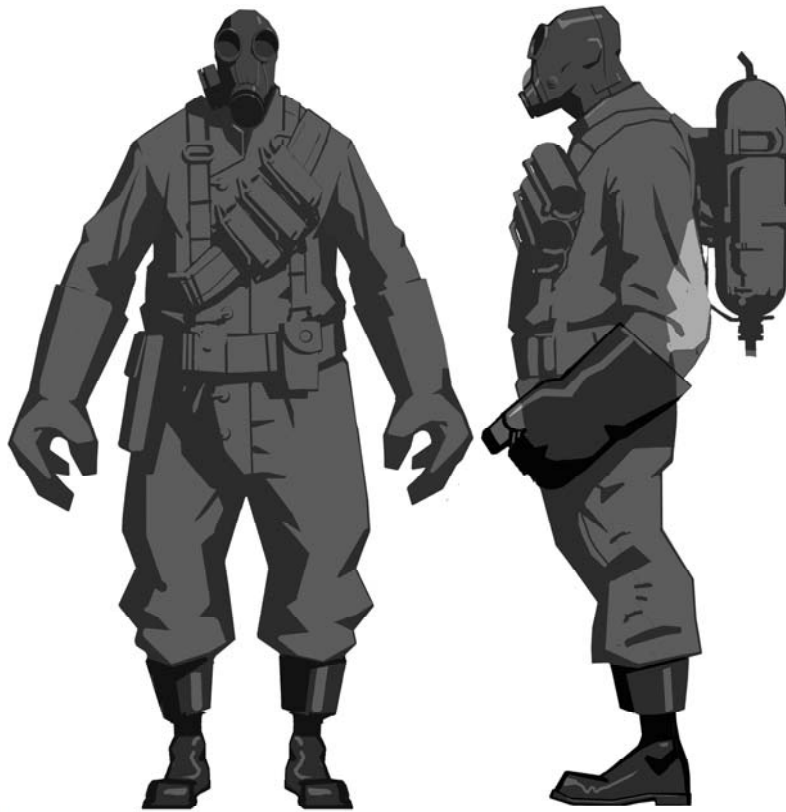
ENGINEER MODEL





TEAM FORTRESS 2

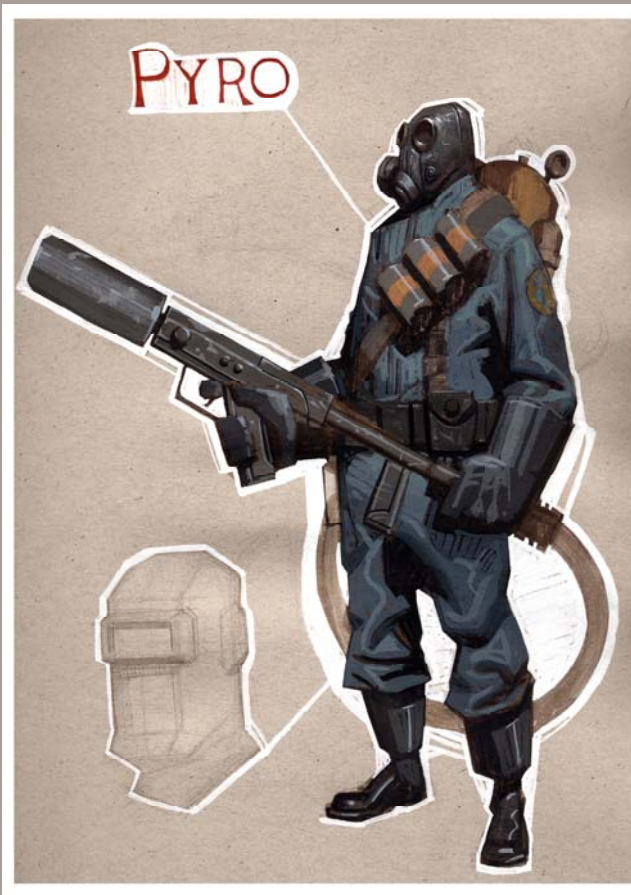
PYRO CONCEPT





TEAM FORTRESS 2

PYRO MODEL





TEAM FORTRESS 2

CHARACTER SHADING ALGORITHM

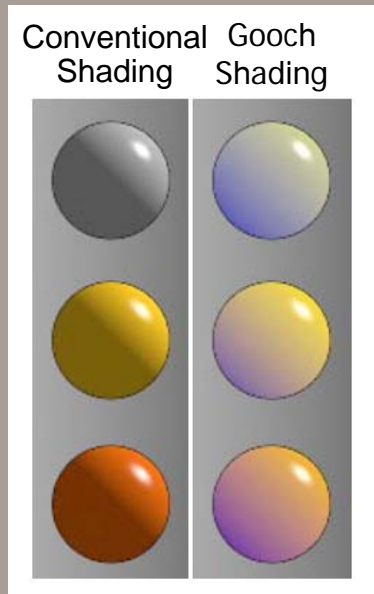
- Previous work in Non-Photorealistic Rendering
- Character lighting equation in *Team Fortress 2*



TEAM FORTRESS 2

GOOCH, 1998

- Hue and luminance shifts indicate surface orientation relative to light
- Blend between warm and cool based upon unclamped Lambertian term, underlying albedo and some free parameters
- Extreme lights and darks are reserved for edge lines and highlights



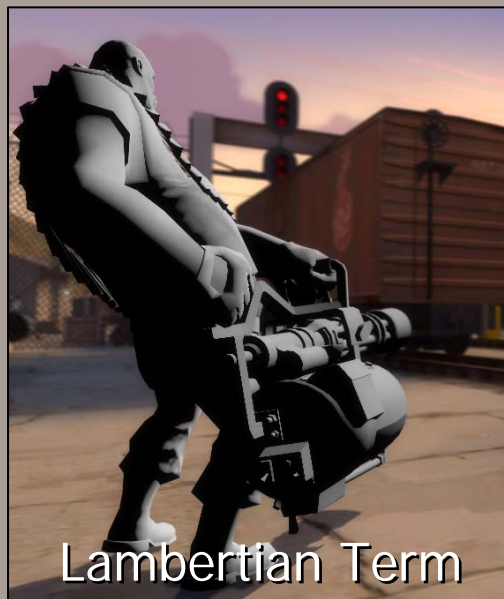
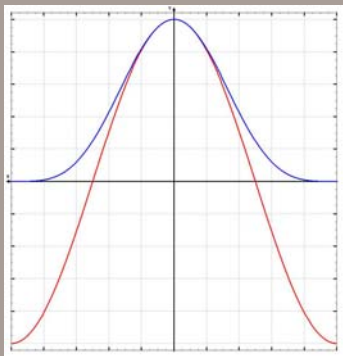
$$\left(\frac{1}{2} (\hat{n} \cdot \hat{l}) + \frac{1}{2} \right) (k_{blue} + \alpha k_d) + \left(1 - \left(\frac{1}{2} (\hat{n} \cdot \hat{l}) + \frac{1}{2} \right) \right) (k_{yellow} + \beta k_d)$$



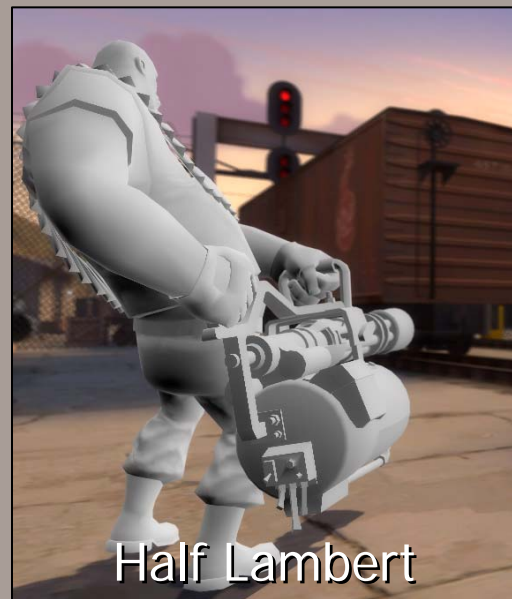
TEAM FORTRESS 2

HALF LAMBERT

- Typically clamp N·L to zero at the terminator
- Half Lambert scales the -1 to 1 cosine term (**red curve**) by $\frac{1}{2}$, biases by $\frac{1}{2}$ and squares to pull the light all the way around (**blue curve**)
- We have been applying this curve since *Half-Life* in 1998
- Similar to *Exaggerated Shading* [Rusinkiewicz06]



Lambertian Term



Half Lambert



TEAM FORTRESS 2

LAKE, 2000

- Lake used a 1D texture lookup based upon the Lambertian term to simulate the limited color palette cartoonists use for painting cels
- Also allows for the inclusion of a view-independent pseudo specular highlight by including a small number of bright texels at the “lit” end of the 1D texture map





TEAM FORTRESS 2

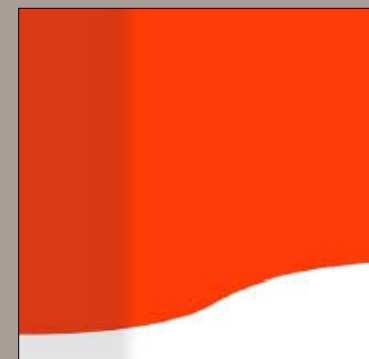
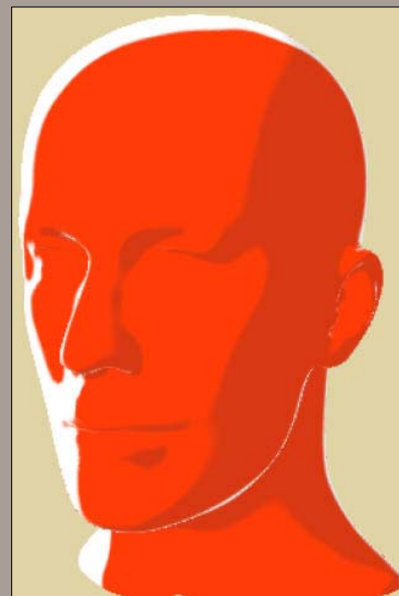
BARLA, 2006

- Barla has extended this technique by using a 2D texture lookup to incorporate view-dependent and level-of-detail effects.
- Fresnel-like creates a hard “virtual backlight” which is essentially a rim-lighting term, though this term is not designed to correspond to any particular lighting environment.



$N \cdot L$

$|N \cdot V|^r$



$N \cdot L$

$|N \cdot V|^r$



TEAM FORTRESS 2

CHARACTER LIGHTING EQUATION

VIEW INDEPENDENT

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right] +$$

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

VIEW-DEPENDENT



TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient





TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms



TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms
 - Unclamped Lambertian term



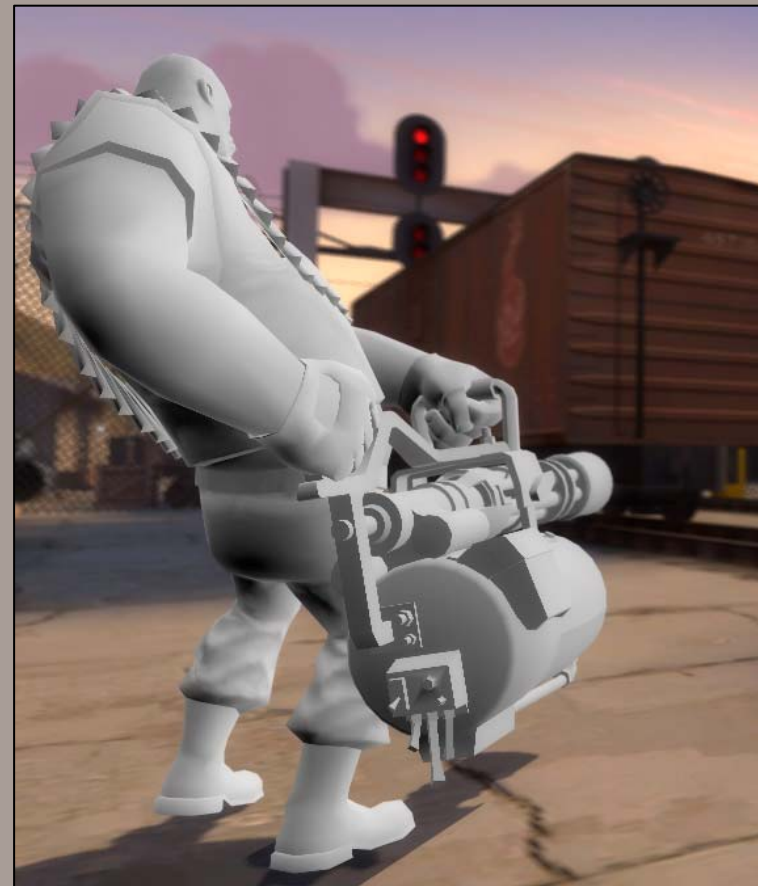


TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms
 - Unclamped Lambertian term
 - Scale, bias and exponent



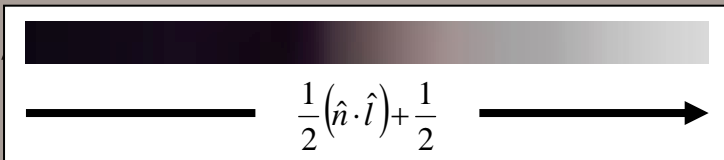


TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms
 - Unclamped Lambertian term
 - Scale, bias and exponent
 - Warping function





TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms
 - Unclamped Lambertian term
 - Scale, bias and exponent
 - Warping function





TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms
 - Unclamped Lambertian term
 - Scale, bias and exponent
 - Warping function
- Albedo





TEAM FORTRESS 2

VIEW INDEPENDENT TERMS

$$k_d \left[a(\hat{n}) + \sum_{i=1}^L c_i w \left(\left(\frac{1}{2} (\hat{n} \cdot \hat{l}_i) + \frac{1}{2} \right) \right) \right]$$

- Spatially-varying directional ambient
- Modified Lambertian terms
 - Unclamped Lambertian term
 - Scale, bias and exponent
 - Warping function
- Albedo





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$



TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light



TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent



TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term



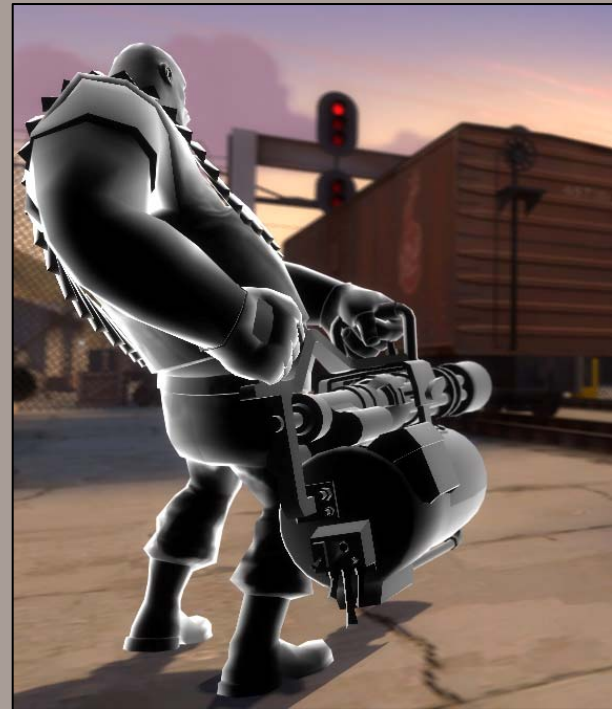


TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture



TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture



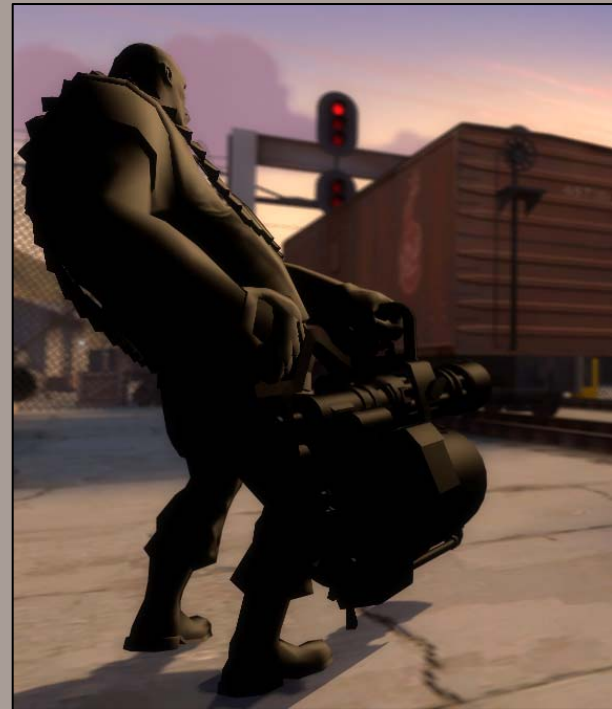


TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture



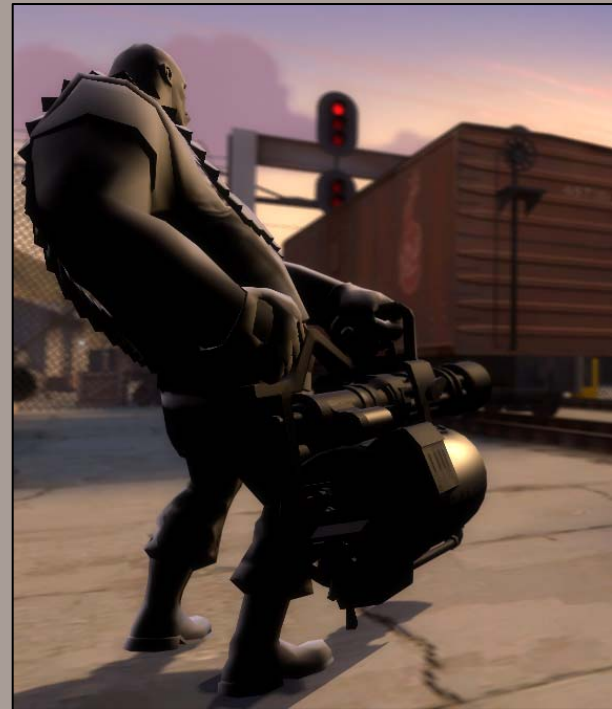


TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v



TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v
 - k_r same rim mask

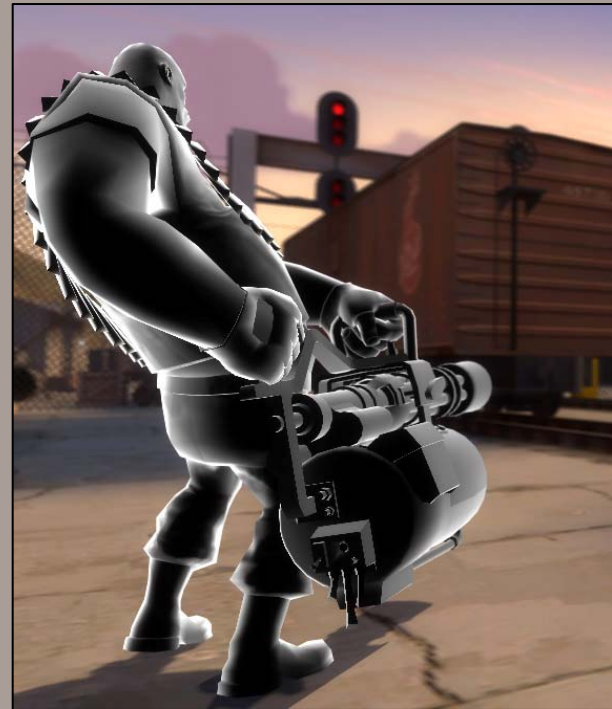


TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v
 - k_r same rim mask
 - f_r same rim Fresnel





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v
 - k_r same rim mask
 - f_r same rim Fresnel
 - $n \cdot u$ term that makes rim highlights tend to come from above (u is up vector)



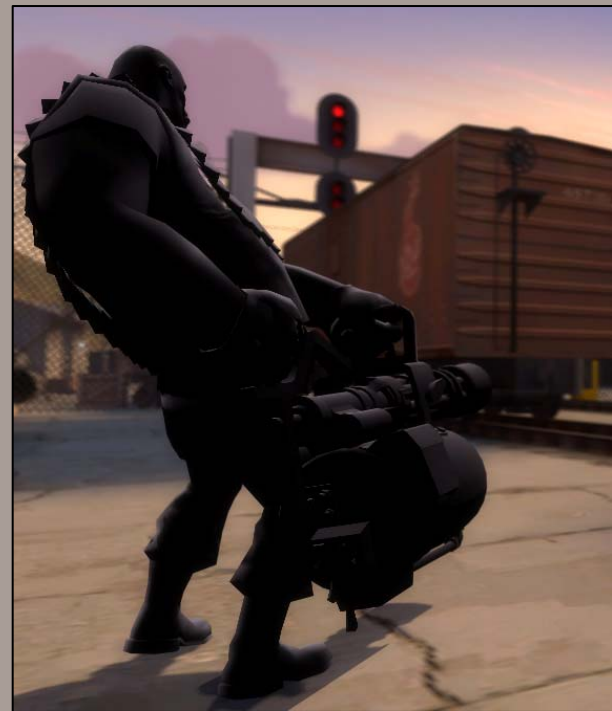


TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v
 - k_r same rim mask
 - f_r same rim Fresnel
 - $n \cdot u$ term that makes rim highlights tend to come from above (u is up vector)



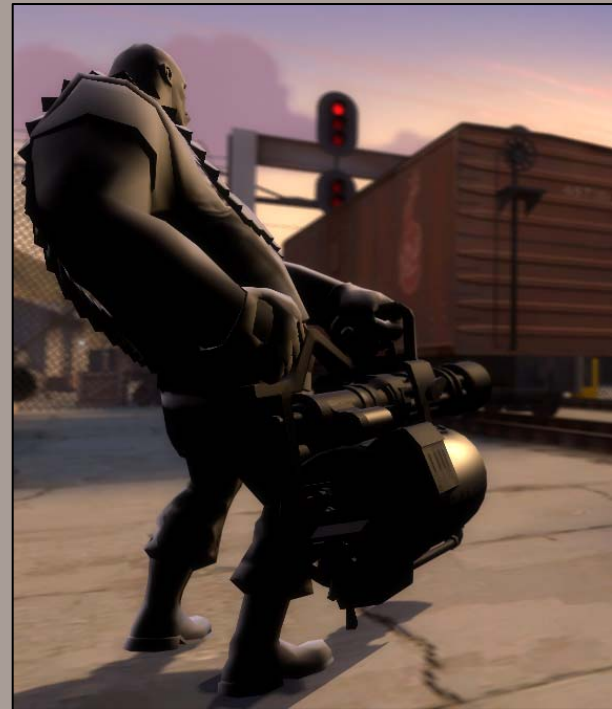


TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v
 - k_r same rim mask
 - f_r same rim Fresnel
 - $n \cdot u$ term that makes rim highlights tend to come from above (u is up vector)





TEAM FORTRESS 2

VIEW-DEPENDENT TERMS

$$\sum_{i=1}^L \left[c_i k_s \max \left(f_s (\hat{v} \cdot \hat{r}_i)^{k_{spec}}, f_r k_r (\hat{v} \cdot \hat{r}_i)^{k_{rim}} \right) \right] + (\hat{n} \cdot \hat{u}) f_r k_r a(\hat{v})$$

- Multiple Phong terms per light
 - k_{rim} broad, constant exponent
 - k_{spec} exponent (constant or texture)
 - f_s artist tuned Fresnel term
 - f_r rim Fresnel term, $(1-(n \cdot v))^4$
 - k_r rim mask texture
 - k_s specular mask texture
- Dedicated rim lighting
 - $a(v)$ Directional ambient evaluated with v
 - k_r same rim mask
 - f_r same rim Fresnel
 - $n \cdot u$ term that makes rim highlights tend to come from above (u is up vector)





TEAM FORTRESS 2

ENVIRONMENT DESIGN

- Creating a compelling, immersive world
- Team distinction through material hue/value/saturation.
- Impressionistic painterly look

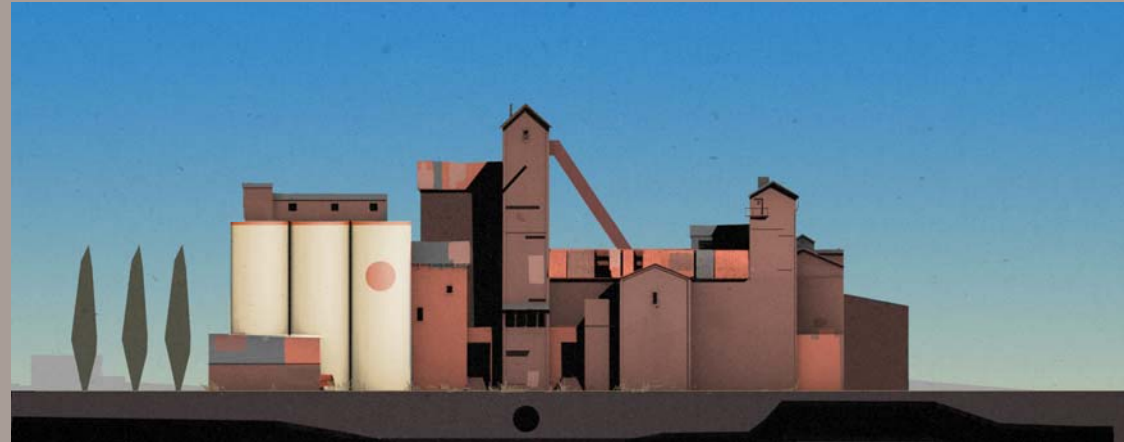




TEAM FORTRESS 2

CONTRASTING TEAM PROPERTIES

- Red
 - Warm colors
 - Natural materials
 - Angular geometry
- Blue
 - Cool colors
 - Industrial materials
 - Orthogonal forms





RED BASE IN 2FORT MAP





TEAM FORTRESS 2

A promotional banner for the video game Team Fortress 2. On the left, a large, muscular character (the Heavy) is shown from the chest up, wearing a red and black uniform and a spiked collar, shouting with his mouth wide open. The background is a bright yellow and orange gradient. The title 'TEAM FORTRESS 2' is written in a large, bold, black, stylized font with a white outline. The letter 'O' in 'FORTRESS' is replaced by a red and white crosshair symbol.





TEAM FORTRESS 2

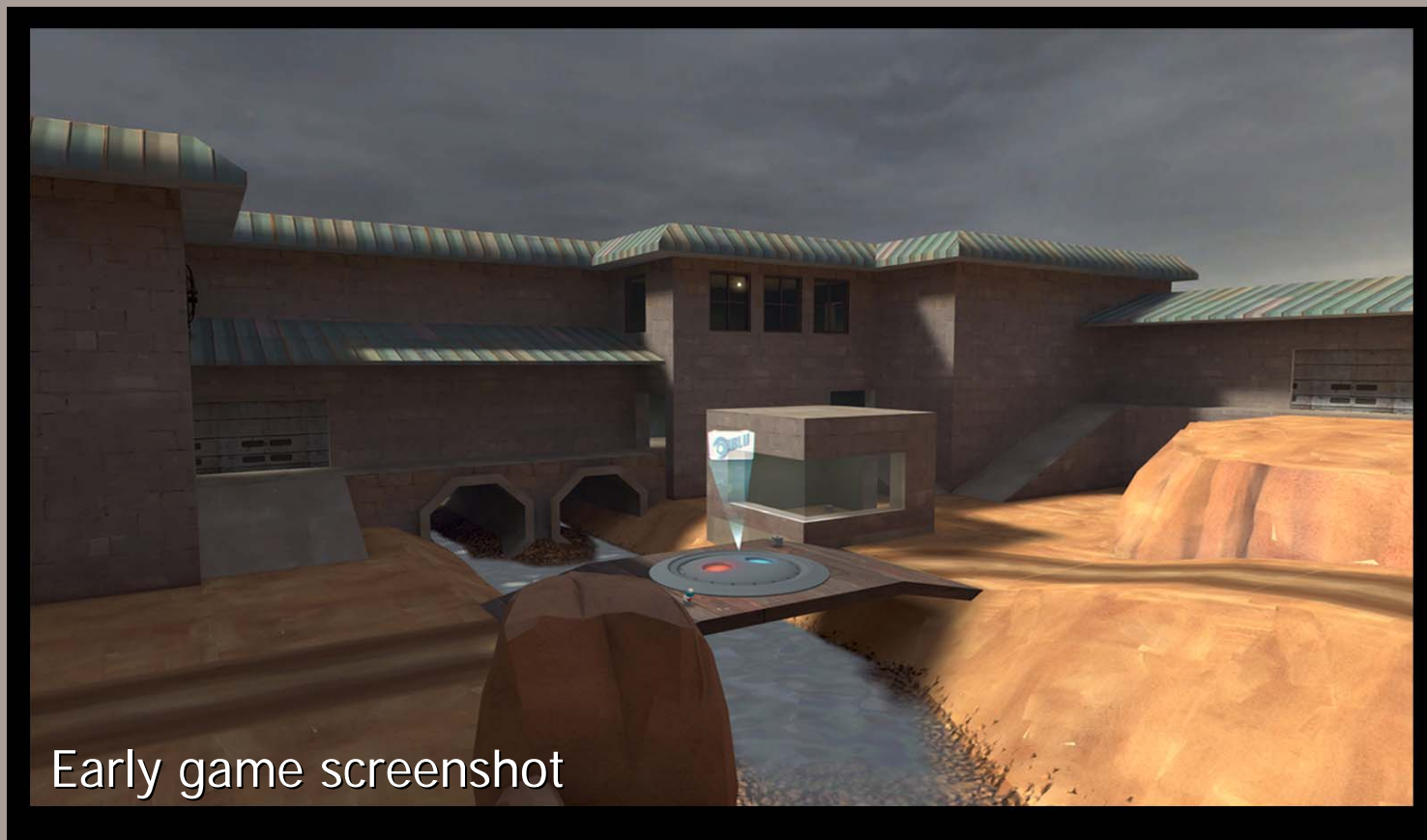
BLUE BASE IN 2 FORT MAP







ROUGH SHELL



Early game screenshot



TEAM FORTRESS 2

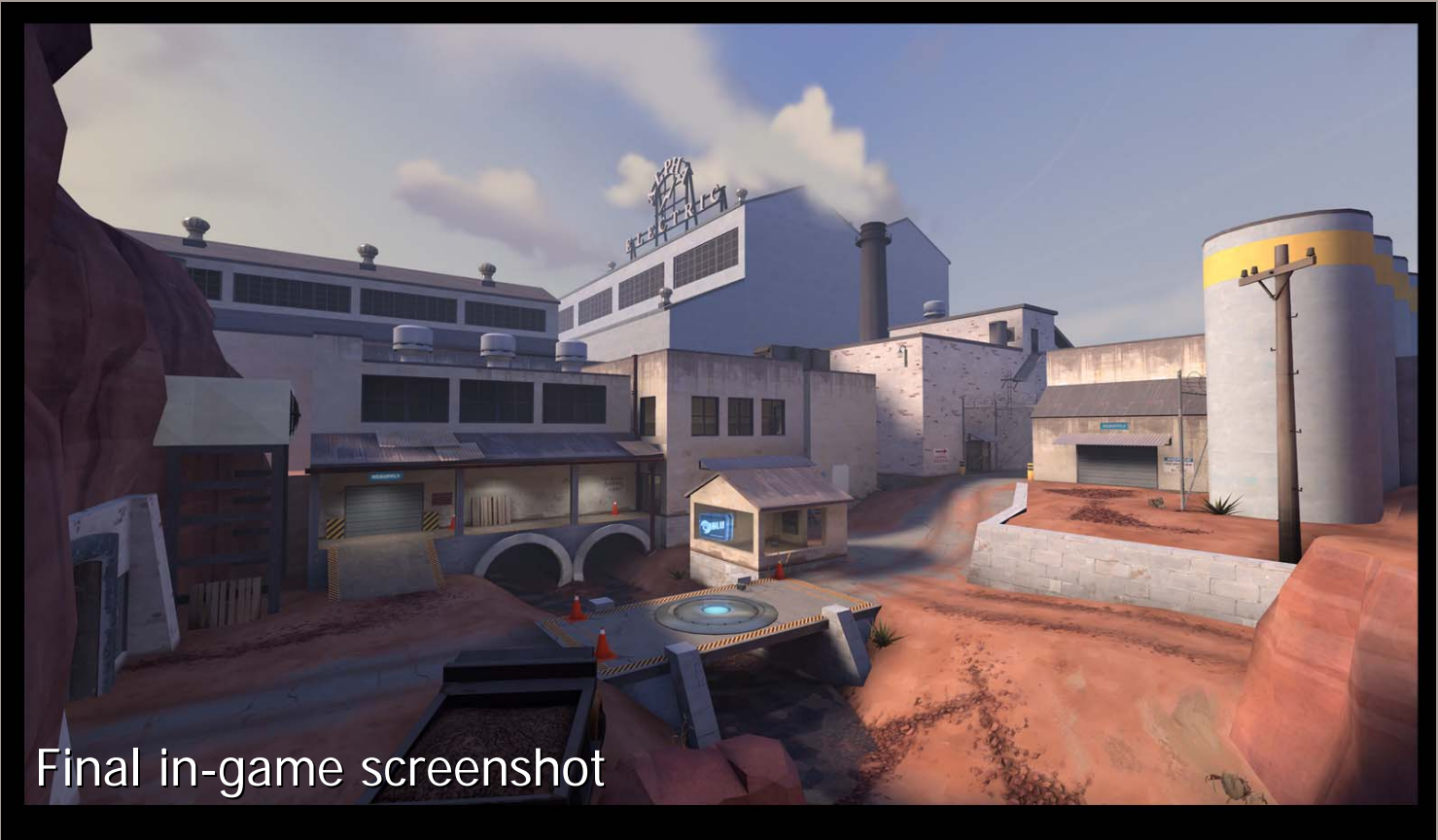
CONCEPT



2D Paintover



ART PASS



Final in-game screenshot



IMPRESSIONISTIC TEXTURES



Shot from *Spirited Away*



TEAM FORTRESS 2

MIYAZAKI - BRUSH WIDTH FORESHORTENED



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





TEAM FORTRESS 2

WORLD TEXTURING



Texture map



In-game Screenshot



WORLD TEXTURING



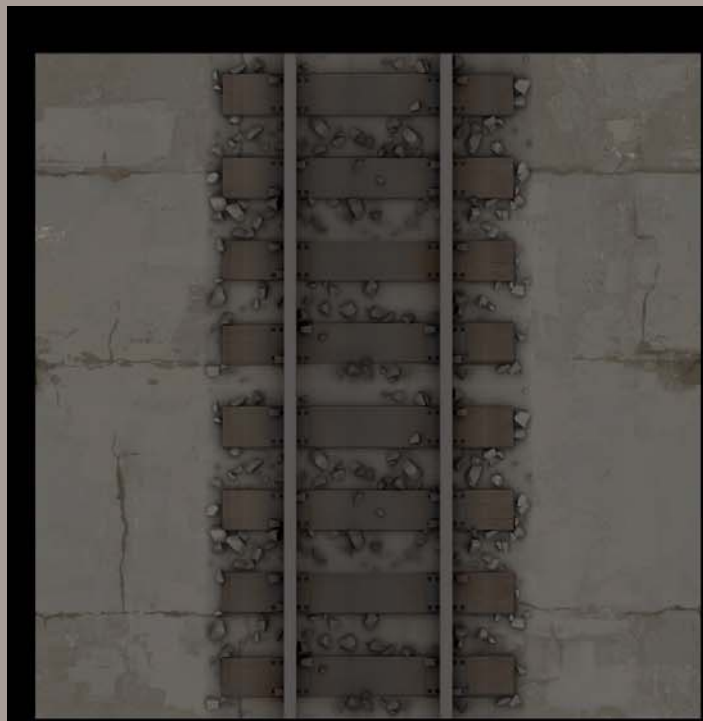
Texture map



In-game Screenshot



WORLD TEXTURING



Texture map



In-game Screenshot



TEAM FORTRESS 2

WORLD TEXTURING



Texture map



In-game Screenshot



MODEL TEXTURING



Texture map



In-game Screenshot



TEAM FORTRESS 2

CLASS = CHARACTER

- Defined personalities and archetypes up front
- Consistent voice casting
- In-game taunt animations and context-sensitive emotes
 - “So much blood...”
- Meet the Team shorts
 - Character vignette movies rendered with game engine
 - Game assets except:
 - Up-rezzed hands
 - More facial morph targets
 - More facial wrinkle maps
- We find ourselves mixing the terms “Class” and “Character”





HOW DID FANS REACT?

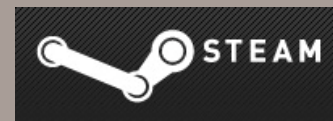




TEAM FORTRESS 2

WHERE DO WE GO FROM HERE?

- Successful multiplayer games live for a long time
- Regular updates via Steam
 - Shipped 28 times since the Beta in September
 - New features, code optimizations and exploit fixes
 - This is why we built Steam & Steamworks in the first place
 - Steam is not just a digital distribution system
 - Can ship updates extremely quickly and fully engage the community
- Extend experience for dedicated players
 - Maps
 - Game modes
 - Achievements
- Unlockable weapons in *Team Fortress 2*
 - Can ship more quickly than new maps and game modes



NEW MEDIGUN CONCEPT



TEAM FORTRESS 2

MEET THE SCOUT

- Things to look for...
 - Distinct character classes
 - Shape and Shading
 - Analogous color palette
 - Painterly world texturing



TEAM FORTRESS 2

CONCLUSION

- History
- Characters
 - Art direction
 - Shading algorithms
- Environments
- Meet the Team
- Post-ship
- *Meet the Scout*





TEAM FORTRESS 2

REFERENCES

- Barla, P., Thollot, J., & Markosian, L. 2006. “X-Toon: An Extended Toon Shader,” *NPAR 2006*
- Gooch, A. A., Gooch, B., Shirley, P., and Cohen, E. “A Non-Photorealistic Lighting Model for Automatic Technical Illustration,” *SIGGRAPH98*.
- Lake, A., Marshall, C., Harris, M., and Blackstein, M. 2000. “Stylized Rendering Techniques for Scalable Real-Time 3D Animation,” *ACM Press, New York, J.-D. Fekete and D. Salesin, Eds.*, 13–20.
- Jason Mitchell, Moby Francke and Dhabih Eng, “Illustrative Rendering in *Team Fortress 2*,” *ACM Symposium on Non-Photorealistic Animation and Rendering*, 2007



TEAM FORTRESS 2

READING LIST

- Art History, Cinematography & Graphic Design
 - *Painting with Light* by John Alton
 - *The Science of Art: Optical Themes in Western Art from Brunelleschi to Seurat* by Martin Kemp
 - *Secret knowledge: Rediscovering the Lost Techniques of the Old Masters* by David Hockney
 - *On Reflection* by Jonathan Miller
 - Anything by Edward Tufte or Marcel Minnaert



TEAM FORTRESS 2

QUESTIONS?

