Over 5.5 million units sold!

Advances in Real-Time Rendering course, Siggraph 2013
Currently working on:

The Witcher 3: Wild Hunt
(PC, Xbox One, PS4, ~2014)
• mature story driven TPP RPG set in a fantasy universe
• jaw-dropping story spanning over 100 hours of gameplay (50h main story arc) that will pull in newcomers and longtime fans alike
• vast, borderless open world with various ways of exploration (sailing, horseback)

Cyberpunk 2077
(PC, next-gen platforms)
• futuristic story driven RPG + open world
• dense city environment + surrounding wastelands
• tons of customization, mixed TPP/FPP

Advances in Real-Time Rendering course, Siggraph 2013
Why characters are important for us?

- storytelling
- believable universe
- expressing and increasing overall emotive response when (and after) playing the game
- our games are story driven
- quality
Character creation pipeline

- lots of information lately on how to render characters
- no information how to get them in the game
- you have to think about animations, AI, mimics, etc…
- how do we get all that information in a consistent way from the content creation tools to the game
Aurora engine (2004 – 2007)

(Bioware - Neverwinter Nights, The Witcher)

- main tool for artists / level designers = 3ds max
- slow iteration times
- characters exported as hierarchy based dummies with one common skeleton containing all the possible animations
- hand painted textures (even normal maps!)
- triangle count: 800 (!)
- character main texture map: 1x 512x512 (!)

Advances in Real-Time Rendering course, Siggraph 2013
The Witcher (2007)
RED engine

• new era, full control
• renderer written from scratch
• it was meant to be used for the Witcher 3 but at the end it was used for Witcher 2
• developed to render characters beautifully
• special shaders, rendering, lighting
• multiple appearances per entity
• underlying technology is pretty simple
• texture skinning with 4 weights per vertex
• animated component using Havok animation
• many hours spent on these special steps
Resources

- moved character setup from 3ds max into the proper engine tools
- export into our own mesh format, extended over the years
- separate resources for textures, meshes, animations, heads
- custom character shaders
- LODs created by hand
- triangle count: ~20k
- typical texture set: 2k head, 2k body (diff, spec, amb, spec2, norm, detail/pattern)
Customization

• modular character design
• vertex collapsing for hats and tight clothes
• embedded tools for patterns / colors / cloth customization
• tools for communities management (spawning NPCs, randomize clothing and coloring)
Initial character production scope - The Witcher 2

• common NPCs: 42, each type have ~5 variants
• main NPCs: 38
• monsters: 17
• main character different armor sets: 15

- $42 \times 5 + 38 + 17 + 15 = 280$

• during Witcher 2 production character team = 5 people!
• pre-production estimations = 27 work days per single character
• $280 \times 27 = 7560$ days = 32.9 years = **6.6 years** of work for 5 artists*

*assuming E3, GDC, and any other marketing request doesn’t even exists!

Advances in Real-Time Rendering course, Siggraph 2013
Modular Design

Face is the most significant part of the human body and it defines identity.
Modular Design

- separate pipeline for the heads
- common topology allows for pose morphing
- face rig setup - motion builder

- teeth + eyes as a separate model

Advances in Real-Time Rendering course, Siggraph 2013
Face rig setup

Advances in Real-Time Rendering course, Siggraph 2013
Modular Design

...and guess what, different heads are not good enough if you want diverse world

Advances in Real-Time Rendering course, Siggraph 2013
Modular Design

- fits the universe
- underlaying history
- best possible quality
Modular Design

- we had to spread the apparent idea to the concept artists
- they've done great job
Modular Design

Advances in Real-Time Rendering course, Siggraph 2013
Shaders

- created visual shader editor
- but only the pixel shader
- generated on the fly
- fast iterations
Shaders

- simple shader albedo shift + brightness
- done in a special shader block
- brute force, additive fresnel for cloth shaders
Geometry components

On average:
- 3 chests
- 4 head covers
- 3 arms
- ...and lots of misc stuff
Recycling

- hi-poly models
- parts of hi-poly models (ropes, sword tilts)
- use the same asset – add a pattern shader
- eventually mixing geometry, patterns and colors gives you a lot of new appearances
Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Rigs & Animation

- hardware texture skinning (GPU)
- using 4x3 matrices (not quaternion), non uniform scaling (dragon)
- 1 vertex 4 bones maximum
- custom pipeline for sound analysis and head/eyes tracking
- using graphs (state machines) for all the animation blending
- supports slots for additional props (bags, weapons, hats, etc.)
- used Havok for animations
- trick for the clothes: rotated bones
- since 2012 we have our own mocap studio!
Level of detail

- special detail level for cutscenes
- most meshes 3 LOD levels + optional billboard
- more detailed
- animations (more bones updated and more sophisticated algorithms)
- meshes
- mimics
RED engine 2

- motion extraction
- loading time issues (appearances, animations, meshes, textures)
- ...and the optical drive seek times
- animation streaming
- events, motion and key pose is always loaded
- streamed animations are always blended in
- tons of optimizations
- background NPCs – lightweight
- triangle count & typical texture set: basically downscaled 2x
CPU usage

- main consumer is the animation
- updates separated into 2 groups
- regular animations and background animations
- background animation are 5 ms sized jobs based on last update time, do as many as possible with priorities
- regular animations are updated in a fork – gather manner on 4 threads (3 jobs + main) with 5 ms budget each
RED engine 3

- dynamic cloth physics
- new mimics system
- appearances dynamically loaded, no need to load all and then choose
- our own animation system
- using just 1 - common rig matrix
- “failed” tessellation attempts
- fur solution
- currently testing PBR + IBL
- triangle count: up to 40k
- typical texture set: 2k for the head, 2k body + 1k misc.
Current character pipeline:
~22 work-days per character for a single artist

- concept art
- sketch model, pose setup
- pre-export (extremely hi poly just to verify that it looks good in the engine)
- retopology (hi-poly split)
- extraction of hi-poly mimical poses (normal blends)
- hi to low poly projection
- texturing (Photoshop – diffuse, specular)
- skinning
- engine export, appearance setup
- shaders, templates, slots, animation behaviors, AI

Advances in Real-Time Rendering course, Siggraph 2013
• character builder tool allows for quick character setup
• character parts are stored based on game specific areas

Advances in Real-Time Rendering course, Siggraph 2013
Generalized normal blend feature

- normal blend component
- the shader has special block that deals with the blending
- 16 areas on a texture
- blend weights per area in constant buffer
- driven by animation component or procedurally

Advances in Real-Time Rendering course, Siggraph 2013
Head pipeline merged

- previously the head component dealt with everything
- now it’s a specialized animated component that deals with the mimics and lipsync
- still needs a bit different handling but much simpler
- mimics have 2 levels, one for general gameplay and one for cutscenes
- tip: animated components can modify the base pose
Tessellation attempts

We tried:

- PN Triangles
  - Doesn’t really add that much detail (characters are already very detailed)
  - It’s hard to deal with the holes

- Displacement mapping
  - Could add more detail to the mesh if the pipeline would be changed
  - Even harder to deal with holes
  - Swimming effect

Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Advances in Real-Time Rendering course, Siggraph 2013
Fur feature

- still in development
- compute shader update for the fur
- pixel shader is customizeable
- fixed PS input structure
- fur component and fur resource to be able to add it separately
- fur is grown from the original mesh
Advances in Real-Time Rendering course, Siggraph 2013
WE’RE HIRING!

Check out positions at our blog: cdpred.com

Advances in Real-Time Rendering course, Siggraph 2013