Principles of Computer Game Design and Implementation

Lecture 30
Principles of Computer Game Design and Implementation

Lecture 30
“Am I a Game Developer Now?”

• Who am I to say?

• We looked at
  – Game architecture
  – 3D game engines
    • including maths required
    • Some physics
  – AI
Learning Outcomes

At the end of the module, the student will have:

1. An understanding of different design issues related to computer games development: game structure, game engine, physics engine;

2. An appreciation of the fundamental concepts associated with game development: game physics, game artificial intelligence, content generation;

3. The ability to implement a simple game using an existing game engine.
Game Architecture

• Game Design
  – Think movies
    • Idea $\rightarrow$ Design $\rightarrow$ lots of work $\rightarrow$ final product

• More arts than technology
  – One can study approaches to design
  – Vast area

• Nothing beats a clear good idea
3D Game Engines

- Graphics
  - Rendering pipeline

http://www.iamthomasvogel.de/?page_id=85
Styled Graphics

• Photo-realistic 3D graphics does not sell
  – ???

• Moody atmospheric graphics
3D Modelling

• We combined geometries within game engine
• 3D Modelling tools
  – Autodesk Maya
  – Autodesk 3ds Max
  – Blender
    • Integration with
Physics

• A tighter integration of physics and game engines
  – Drawing fur, grass, etc
  – Particles
  – Flame
  – ...

Animation in Games

• We modelled object motion
  – a kind of animations

• Characters should move realistically
  – Modelled in a 3D modelling tool (blender)
  – Provide “hooks” to play sequence from game

• Motion capture
  – Play the sequence
Keyframe Animation

- Storing (and processing) each frame is too expensive
- Keyframe animation: store a (relatively small) number of keyframes and *interpolate*
Animation of Models

- Rigid body animation
  - Body is immutable
  - Sequence of keyframes

- Skeletal animation
  - Bones
  - Skin
    - Follows the skeleton
Inverse Kinematics

• Normally, animation is **forward kinematics**
  – Sequence of keyframes specifying bone motion

• Inverse kinematics
  – Specify where you want a bone to move
  – Animate the model
    • Pick up an object
  – Limits have to be set!
Content Generation

• Modern games are (by in large) about assets
  – Worlds to explore
  – Enemies to kill
  – Friends to make

• Level designers
Procedural Content Generation (1)

• Assets generated by an algorithm
  – As a tool for game developers

www.speedtree.com
Procedural Content Generation (2)

- Terragen
  
  http://planetside.co.uk
Procedural Content Generation (3)

• Assets generated by an algorithm *on the fly*
• Map generation
  – Dungeon generation in 2D
  – Problems with 3d
    • Too slow
    • Too dull
    • Verification required
Example: A Growing Tree Algorithm

2D maze generation
• Pick a maze cell
• See if there’s space to grow into
  – Random direction
• Carve into the space
• Repeat until finished
Procedural Content Generation (3)

• Assets *tuned* by an algorithm
• Face Instances

• Borderlands
  – Combinations of guns
• Spore
  – Combinations of features
Procedural Content Generation (4)

Procedural population

• S.T.A.L.K.E.R.: Shadow of Chernobyl
  – Dynamical placement of characters
    • Artificial Life

• Left4Dead
  – In addition to placement, adaptive pacing
  – If intensity is too high, remove major threats for a while
Conclusion

• These are just some of directions
• Lots of further info online
  – www.gamasutra.com
  – aigamedev.com
  – www.gamedev.net
  – ...
• Tons of books
• Experiment yourself!