

CSE 481D
Software System Design
Autumn 2008

John Zahorjan
Deepak Verma

Today's Agenda

- Course Overview
 - What are the goals?
 - What will happen? / What's expected of you?
 - How are grades determined?
 - ...
- Form teams
 - 3 teams of 5 people
- Game Design Overview
- First "assignment"

Course Overview: Capstones

- This is a capstone course
 - About doing more than listening
 - Problems not compartmentalized into neat subject areas
 - Open ended projects
 - You have authority / responsibility over almost all decisions
 - Experiences not available in traditional courses, in grad school, nor as a new hire
 - "Freedom to fail"
 - A march, not a sprint

CSE 481D

Software Systems Design: Games

- The emphasis here is on the process of designing, building, and deploying software
 - The process trumps the product
 - Aim high!
- We build games
 - Why? They're fun!
- One (big) lesson you'll observe:
 - Success is dependent on a lot more than coding skill

Activities Overview

- We'll form into teams of about five
- Each team will create an initial game design
 - The design must include something for everyone
 - Compromise inevitably means there are things you would do differently
- Make some implementation decisions
 - What OS / language / compiler / source control?
 - Choose game engine (rendering, sound, networking, physics, ...)
 - Choose content production pipeline
- Choose a development process; assign roles
- Establish working environment on machines

Activities Overview (cont.)

- Initial development in parallel
- "Integration" due 4 weeks from today
 - Integration is gluing of separate pieces of code into a single code base
 - Ideally, at this point you have a crude version of your game
- "Beta release" 4 weeks after that
 - You'll each play the games of the two other teams and provide feedback to them
- Public demo of games 9 days after feedback is due
- Short final report due middle of exam week

A bit of explanation

- This activity isn't just coding up a spec
 - It's creating the initial spec
 - updating the spec as you go along
 - monitoring progress and acknowledging reality
 - many other things besides coding...
 - *It's doing the best you can with the finite resources you have*
- New this year:
 - the designer, the implementor, and the user often have very different ideas about how things should behave, and what's an important feature and what isn't

Let's look at the detailed schedule

- <http://www.cs.washington.edu/education/courses/cs>

Class Responsibilities

- Individual students
 - Major responsibility is to your team
 - Reliable is as important (maybe more important than) fast
 - Also responsible to yourself:
 - There's an opportunity (but not an obligation) to do things you don't know how to do, or maybe don't think you like to do
- Teams
 - Must communicate with "management"
- Instructor / TA
 - Offer advice about both technical and non-technical issues
 - Resolve problems, technical and non-technical
 - Perform final evaluation

Grading

- I won't lie, it's tough to assign grades
- Criteria:
 - How much did you contribute (positively) to your team's success?
 - There are many ways to contribute. This isn't just a matter of counting lines of code.
 - How well did your team do as a whole?
 - It's everyone's fault if the team burns up vastly more resources than the final product "should have taken"
- Input to this process:
 - Meetings with teams and individuals
 - Observations, including looking at the source
 - Final report

End of Section I

- Things I didn't mention?
- Other questions?

Let's Form Teams

- Many ways to do this
- Experience says that one good way is to try to cluster by when you can meet/work
 - Mornings
 - Afternoons
 - Evenings
- (Aside: Working together, meaning working in the lab, is a huge, huge advantage.)

Game Design

- Your first task as a team is to create an initial game design
- The game should be:
 - Fun
 - Buildable in about 8 weeks
- Neither of these is an easy thing to evaluate
- We'll talk about building next Thursday...
- Today, a brief overview of "fun"

Game Design

- There is little (useful) formalization, or even informal agreement about, what makes a good game
- However, good designers definitely understand things that most of us don't
 - Our own intuition about why we liked some game often misses the essential
- If you're deeply interested in game design, my recommendation is to look at <http://www.gamasutra.com/>
- I'll summarize two articles now
- Please read the full articles *before* your team starts its game design!

Article 1: Emotional Involvement

- "Why We Play Games: Four Keys to More Emotion Without Story," Nicole Lazzaro, XEODesign.com.
 - Available at <http://xeodesign.com/whyweplaygames.html>
- Studied three groups of people: hard core gamers, casual gamers, and non-gamers.
- Wanted to understand how to inject emotional responses into games without story cut-scenes

Essential Result



Criteria for 4 Keys

1. What Players Like Most About Playing
2. Creates Unique Emotion Without Story
3. Already Present in Ultra Popular Games
4. Supported by Psychology Theory and Other Larger Studies

- Successful games engaged players with at least 3 of these 4 "keys"

Elicited Emotions

Emotion	Common Themes and Triggers
Fear	Threat of harm, object moving quickly to hit player, sudden fall or loss of support, possibility of pain
Surprise	Sudden change. Briefest of all emotions, does not feel good or bad, after interpreting event this emotion merges into fear, relief, etc.
Disgust	Rejection as food or outside norms The strongest triggers are body products such as feces, vomit, urine, mucus, saliva, and blood.
Naches/ Kvell (Yiddish)	Pleasure or pride at the accomplishment of a child or mentee. (Kvell is how it feels to express this pride in one's child or mentee to others.)
Fiero (Italian)	Personal triumph over adversity. The ultimate Game Emotion Overcoming difficult obstacles players raise their arms over their heads. They do not need to experience anger prior to success, but it does require effort.
Schadenfreude (German)	Gloat over misfortune of a rival Competitive players enjoy beating each other especially a long-term rival. Boasts are made about player prowess and ranking.
Wonder realm	Overwhelming improbability. Curious items amaze players at their unusualness, unlikelyhood, and improbability without breaking out of of possibilities.

Game Characteristics

- "Hard Fun"
- It's about winning
 - *Playing to see how good I really am*
 - *Playing to beat the game*
 - *Having multiple objectives*
 - *Requiring strategy rather than luck*

Game Characteristics

- "Easy Fun"
- It's about exploration (not winning)
 - *Exploring new worlds with intriguing people*
 - *Excitement and adventure*
 - *Wanting to figure it out*
 - *Seeing what happens in the story, even if I have to use a walk through*
 - *Feeling like me and my character are one*
 - *Liking the sound of cards shuffling*
 - *Growing dragons*

Game Characteristics

- "Altered States"
- Games as therapy
 - *Clearing my mind by clearing a level*
 - *Feeling better about myself*
 - *Avoiding boredom*
 - *Being better at something that matters*

Game Characteristics

- "The People Factor"
- Socialization
 - It's the people that are addictive not the game.
 - I want an excuse to invite my friends over.
 - I don't like playing games, but it's a fun way to spend time with my friends.
 - I don't play, but it's fun to watch.

Article 2: Interview with a Game Design Artist

- "Game Design Psychology: The Full Hirokazu Yasuhara Interview",
http://www.gamasutra.com/view/feature/3769/game_design_psychology_the_full_.php
- Mr. Sonic Hedgehog

How do your ideas come from these individual moments into the full art of game design?

- HY: *So I always think about all the different elements of what makes something fun. This formula is made by a sociologist from France who did some thinking into what it is that makes something fun, or interesting, for people to experience. One of the things is competition. The next is happy coincidences; a gamble that pays off, that kind of thing. Following that is dizziness or exhilaration, and the final thing is imitating, or copying.*

(cont.)

- *Another important thing is to consider the basic desires of people, even if all you're thinking about is a simple game. For example, you have active desires -- "Freedom from Fear", as they say, the way people actively want to avoid fear in their lives. And one way they deal with that is by engaging in a sorting process.*
- He's talking about game play as imposing order, something people like.

Varying Time-Scales of Objectives

- *Another thing is that, putting it simply here, I usually come up with three goals when I'm making a level: a short-distance, middle-distance, and long-distance goal.*

For example, going to see Cinderella Castle in Disneyland would, for us, be a long-distance goal; if it was a game, we'd need to keep reminding the player where he's going if we actually want him to remember it.

A more short-distance goal, meanwhile, would be if you're in a baseball game; your goal is to get on base, and there are any number of simple, linear ways to achieve that goal. An example of a middle-distance goal would be if you run into a bridge in the forest that you can't gain access to -- something I do a lot in games. Maybe you have to do a sequence of jumps to reach it, but it's visible, at least.

So in your opinion, how has design changed from 2D into 3D?

- *I think it's mainly in the camera -- its positioning. That can change everything. If you place the camera to the side, then it's a "2D" game, but in a 3D game, it's all in how well you can express the world to the player, how clearly you can show elements and obstacles. There are lots of approaches to solving that problem, so there are lots more possibilities to explore.*

...

I definitely spent a lot of time thinking about the camera -- whether it was too close or too far. If it's too far, then you'd start to have polygon issues, but if I put the camera down lower, then you couldn't see far enough ahead. So I experimented with raising the camera when there weren't too many polygons on screen, and so forth. It was a major headache for me.

I also heard that originally, when Sonic would get hit, the rings would not come out. Is that true? And then that was later implemented to make it more interesting.

- *Well, I think the way rings shot out of you when you got hit was there from the beginning. But the Genesis's power means that you can only show so many rings at once, so we experimented a lot at first with having the rings flash and not overlap with each other and so on.*

After a while, though, we realized that having a ton of rings onscreen would be a selling point -- it'd show how cool the Genesis was, and it was visually interesting. So we tried our hardest to make that happen.

Assignment For Next Tuesday

- Create a game design document describing the essentials of your game
 - Overall narrative / goal
 - Key player activities
 - Strategy alternatives offered to players
 - Pacing: short, medium, and long term goal(s)
 - Opportunities to tweak difficulty / tune fun factor
 - At least two sketches of what the screen might look like
 - What MUST you get done for the game to be fun?
- Put the GDD in the course wiki
- Present the game during Tuesday's class

A Few Observations

- Try to avoid a game for which the key element is something we're not sure we can build:
 - Examples: highly intelligent AI; extremely high performance; sophisticated animations; lots of content (e.g., many levels)
- What we're sure we can build: First Person Shooters (FPS)
 - We've also had third person games, but camera control is HARD. The game needs to be designed with the camera in mind.
 - RPGs haven't worked out...

Useful Links

- XEODesign article:
<http://xeodesign.com/whyweplaygames.html>
- Yasuhara interview:
http://www.gamasutra.com/view/feature/3769/game_design_psychology_the_full_.php
- Understanding the visuals:
http://www.gamasutra.com/view/feature/3753/the_art_of_braid_creating_a_.php
- A gaming lexicon:
<http://www.half-real.net/dictionary/>