Valve’s Design Process for Creating Half-Life 2

Presented by David Speyrer and Brian Jacobson
The Fuzzy Problem of “Fun”

- Obvious in hindsight — “I know it when I see it”
- Has many solutions
- Subjective
- Defies direct analysis
Define your goals and constraints
Come up with an idea of how to meet them
Perform an experiment to test the idea
Evaluate the quality of the experiment
Evaluate the quality of the idea
Evaluate the quality of your goals
Repeat
Necessary Ingredients

➤ The right attitude
➤ Well defined, measurable goals
➤ Well communicated goals
  • Niche product?
  • Mass market?
➤ Well devised tests
Defining Goals

- “Product focus” helps you define good goals
- Care more about the quality of the product than your particular contribution to it
- Filter all goals through the lens of customer experience
- Good customer experience equals success
Goal is a fun game
Ideas are your game designs
Playtests are your experiments
Evaluate your designs as a result of playtests
Repeat
What does “playtest” mean?

- QA?
- Balancing?
- Focus testing?
- Fun?
Running a Good Playtest

- Are playtesters having the experience you designed?
- Is the experience you designed desirable?
- Learn about things that affect customer experience
  - Game code/NPC behavior
  - Effects art
  - Environmental art
  - Sound
  - Training
  - Pacing
  - Difficulty

© 2007 Valve Corporation. All Rights Reserved.
Make sure the people responsible for the design and execution are there
- Simplifies evaluation
- Prioritizes
- Motivates

Simulate the player “in their living room”
- Don’t give them hints
- Don’t answer any questions
- Don’t provide extrinsic rewards

Use external playtesters
Questioning Playtesters

- Don’t rely too much on questions
- Often you learn more from what playtesters don’t experience
- Ask non-leading questions
- Can be great for measuring effectiveness of certain elements
  - Storytelling
  - Perception
Design Iteration

➢ Often this occurs late in production
  • Some of your designs work, others don’t
  • Fix the most egregious problems

➢ Late playtesting is less valuable
  • It’s too late to make substantive changes
Playtesting as Production

- Use playtest results to drive production!
  - Create 15 minutes of gameplay in rough form
  - Playtest
  - Use playtest to prioritize work for next week
  - Repeat until complete

- We felt done as soon as playtesting was no longer painful to watch
Small Increments

Do the smallest amount that lets you learn something about the player experience

Use 1-2 week increments
- Shorter results in not enough time to make changes
- Longer results in churn and flail
“I’m Just Worried That...”

- Don’t let theoretical problems prevent playtesting
  - They might not actually be problems
  - If they **are** problems, the playtest will prioritize which to solve first
  - Playtest may generate ideas of how to solve actual problems better

- Don’t worry about how it looks
  - Art production is less risky than gameplay production
Other Benefits

- Useful for learning
- Easy to measure an element’s incremental value or damage
- A great way to avoid design arguments
- Can use playtest results to drive other aspects of production
Playtesting as Production

- Solutions to playtest problems can be iterative
- Solve your problems in the right order
- Look for trends
  - Don’t overcorrect
  - Don’t oscillate
- Finish successful elements before moving on
Product-level Benefits

- Allows you to schedule to a particular quality metric
- Scopes game design risk for key features
- Allows you to optimize toward your most successful elements
- Allows you to measure risk, speed, cost
Playtesting as Production in Larger Projects

- Create multiple small independent design teams
  - Each chapter was done by a particular design team
- Create a sandbox for each team to work in
- Create processes to help with global decisions
  - Story
  - Global mechanics (weapons, NPCs)
  - Art
  - Consistency
  - Quality
Process #1: Establish Initial Constraints

- A preproduction phase established initial product decisions
  - Story elements and settings
  - Art concepts/style guides
  - Major design principles
  - NPCs, mechanics, weapons, vehicles
  - Chapter progression and themes

- Prototype gameplay maps were created
Process #1: Establish Initial Constraints

- Some decisions were used by design teams as constraints
  - Story, settings, design principles

- Others were treated as suggestions
  - Mechanics, weapons, enemy NPCs were picked up by design teams
  - Some elements never were adopted

- Some major elements in the shipping game were developed after this phase
Process #2: Promote Design Economy

- Encouraged reuse of existing game elements in new ways
- Useful in helping with global consistency and quality
  - More of your game is about the same elements
  - More hands working on each element improves quality
- Used teamwide playtests to expose elements to other design teams
  - Successful elements naturally diffused through the game

+ = FUN
Process #3: Establish Strike Teams

- Formed to address cross-team issues
- Some strike teams existed for the entire project
  - The “Weapons Cabal”
- Most were more transient
  - Occurred when a design team used another’s gameplay elements
- Decisions in well-tested maps were treated as constraints
Process #4: The “Overwatch Cabal”

- Evaluated global product-level quality at Alpha
- Communicated high/lows to all design teams
  - List constructed from company-wide feedback
- Consisted of a member from each design team and art/sound/animation teams
- Design teams were responsible for addressing feedback
  - Cuts/changes were driven by individual teams
  - All changes were made during the Alpha period
What kind of changes should you make during Alpha?

- Don’t introduce major new elements
- Be ruthless and cut your worst problems
- Do add density if necessary using existing elements

Some aspects of your game can’t be measured until it’s all there

- Pacing
- Difficulty curve
- Variety
- Chapter-to-chapter inconsistencies
Conclusions

- Engineering process can be applied to game design
- Let your production teams drive your design
- Use playtesting to drive game production
- Large teams can use this technique if the appropriate processes are in place
- Allow for a final iteration over your entire game once it’s playable from beginning to end