
# Sound Practices of Games Business and Design

Presented by **Brian Jacobson**

VALVE

# "Soft" Problems for Games Businesses

- Game design
- Storytelling
- Marketing
- Customer experience

VALVE

## The Engineering Approach

- Define your goals
- 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

VALVE

## Necessary Ingredients

- The right attitude
- The right people
- Well-defined goals
- Well-communicated goals
- Well devised tests

VALVE

## Setting Goals

- "Product focus" helps you define good goals
- Filter all goals through the lens of customer experience
- Seek helpful constraints
  - Don't try to solve boundless problems
- The "hardcore gamer" problem

VALVE

## Engineering Game Design

- Goal is a fun game
- Ideas are your game designs
- Playtests are your experiments
- Evaluate your designs as a result of playtests

VALVE

## What does “playtest” mean?

- QA?
- Balancing?
- Focus testing?
- Fun?

VALVE

## 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

VALVE

## Running a Good Playtest

- Make sure the people responsible for the design and execution are there
  - Simplifies evaluation
  - Prioritizes
  - Motivates
- Don’t gather stats!
- Use external playtesters
- Don’t say anything to playtesters
- Ask playtesters to speak through what they are thinking while they play

VALVE

## Questioning Playtesters

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

VALVE

## Design Iteration

- Oftentimes this occurs late in production
  - Some of your designs work, others don't
  - Fix the most egregious problems
- The "legendary" designer
  - A designer whose designs always work
  - We have no such designers at Valve

VALVE

## 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

VALVE

# Playtesting as Production (Half-Life 2)

VALVE

## Small Increments

- Do the smallest amount that lets you learn something about the player experience
- Use 1-2 week increments
  - Less results in not enough time to make changes
  - More results in churn and flail
- Build about a few hours of game, then start again
  - We felt done as soon as playtesting was no longer painful to watch

VALVE

## Tech development

- Options
  - Build a new engine
  - Build off your previous title's engine
  - Use a licensed engine
- You do need to do some up-front work
  - But not too much -- this was our big mistake on Half-Life 2
- Use iterative tech development
  - Identify key technology bets – do those first

VALVE

## In Theory

- 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 discard game designs on theoretical problems

VALVE

## Other Benefits

- Useful for idea generation and learning
- Easy to measure an element's incremental value or damage
- A great way to avoid design argument
- Can use playtest results to drive other aspects of production

VALVE

## Playtesting as Production

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

VALVE

## 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

VALVE

## Game Design Experiments

- Use them in your games!

VALVE

## Engineering Customer Experience

- What's most important to our customers?
- Is our marketing effective at reaching our customers?
- What are the worst problems plaguing our customers?
- What perf and memory budgets do we need to meet?
- Steam

VALVE

# Steam

- Building online games
- Building features for the customer
  - Auto updates
  - Anti-cheat
  - Communications – e.g., Friends
- Building features for your business
  - Product encryption/anti-piracy
  - Direct sales
  - Measurement

VALVE

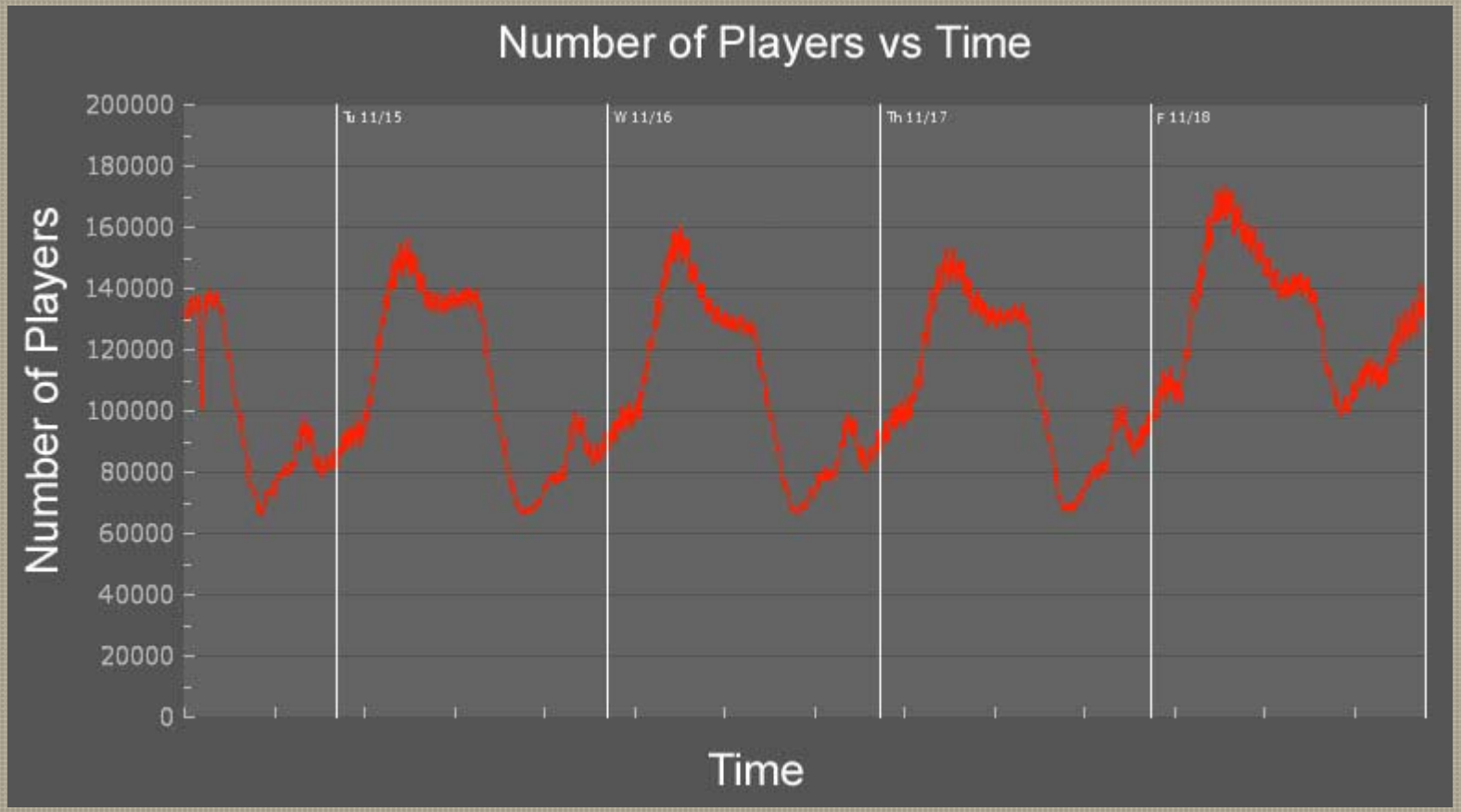
# Target Platform Experiments

- www.steampowered.com/status/survey.html

| Video Card Description | | | |
|---|---|---|---|
| ATI Radeon 9600 Series | 79,818 | 9.55 % | |
| ATI Radeon 9800 Series | 78,006 | 9.33 % | |
| NVidia GeForce FX 5200 Series | 66,756 | 7.98 % | |
| NVidia GeForce4 MX Series | 53,963 | 6.45 % | |
| NVIDIA GeForce 6600 | 45,801 | 5.48 % | |
| ATI Radeon 9200 Series | 42,291 | 5.06 % | |
| NVidia GeForce4 Series | 40,443 | 4.84 % | |
| NVidia GeForce2 MX Series | 27,297 | 3.26 % | |
| NVidia GeForce 6800 GT | 22,743 | 2.72 % | |
| NVidia GeForce FX 5600 Series | 20,511 | 2.45 % | |

VALVE

# Product Success Experiments



VALVE

## Other Experiments

- Number of Steam sales/pre-orders
- Number of registrations (online and retail)
- Number of crashes

VALVE

## Conclusions

- Use the experimental approach today!
- Use playtesting to drive game production
- Steam is one tool that can help