

# Video Game Design

Jeff Wilson

[jeff@imtc.gatech.edu](mailto:jeff@imtc.gatech.edu)

Prototyping



# Why build prototypes?

- Building game without a prototypes like “...shooting a movie without a script”
- Rough approximation of game
  - Physical
  - software
- No complex algorithms, media content etc.
- Avoid production related issues
- Informal content conducive to radical changes



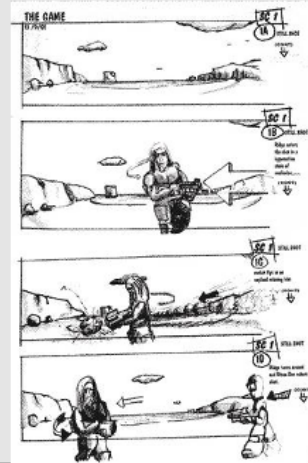
# Role of Prototype

- Test
  - Game mechanics
  - Balance of rules
  - User experience
- Discover play patterns and emergent behavior
  - e.g “Whoever gets the shotgun first, wins.”
  - ...”A successful player always forms alliances early in the game”
  - ...”spawn camping”
- Almost impossible to just “figure it out in your head”



# Prototyping Techniques

- Paper prototypes
  - Work well for testing game mechanics, rules, procedures
    - Many video games come from board games
  - Very quick to produce
  - Hard to capture action or experience of game
    - Some games more appropriate than others
- Video, Storyboards/Animatics
  - Captures user experience
  - Video can be tedious to produce and change
  - Useful for communicating ideas to others



<http://video.google.com/videoplay?docid=-5994464494334668806&q=game+animatic&hl=en>

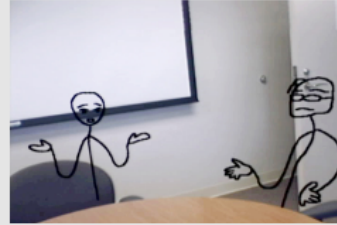
# Prototyping Techniques

- Wizard of Oz
  - Useful to simulate complex behavior
    - e.g AI or new types of interfaces
  - But may need to simulate real issues (e.g. network latency) too
- Rapid software development
  - Flash, Director, VB, Level Editors, Modern Game Engines, etc.
  - Can combine with paper techniques



# Lessons Learned

- AR Experience “Four Angry Men”
- 6 months of content development...three times
- 1 year of software development before testing was possible
- Motivated creation of Designers’ Augmented Reality Toolkit (DART)

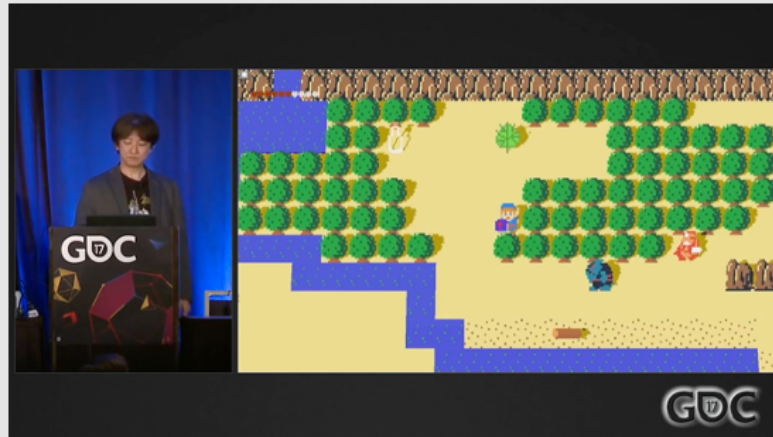


# Choosing a Technique

- What aspects of game need prototype testing?
  - e.g. paper to start with, hybrid approach as complexity increases
- What are the requirements of the prototype?
  - e.g. visuals not needed, number crunching required
- What is the quickest way to build the prototype?
  - **Use tools you know**
  - **Avoid overkill and unnecessary complexity**
- Will the prototype be flexible enough?
  - What elements will need to change and how often
  - Be cautious of reusing code in final game



# Zelda - Breath of the Wild



Full talk: <https://www.youtube.com/watch?v=QyMsF31NdNc>



# Dangers of Software Tools

- Every prototype ends up sharing lots of common elements (e.g. default graphic elements, widgets, etc.)
- Difficult to explore new interactions without bias



# The Steps in Prototyping

- **Begin with core gameplay (progression of action)**
  - Most common game mechanism for player
    - “Super Mario Bros.: A player controls Mario (or Luigi), making him walk, run, and jump, while avoiding traps, overcoming obstacles, and gathering treasure”
  - Start with core and build outwards



# The Steps in Prototyping: Step 1

- Foundation
  - Basic objects and key system procedures
  - Essential rules
  - e.g. FPS player can do actions each turn: move, turn, shoot
  - Don't tackle complex questions yet
    - “how do the shields work?”,
    - “What if the player can fly?”



## The Steps in Prototyping: Step 2

- Structure
  - Elements that support the foundation
  - Those most essential to the game
    - e.g. number of spaces a unit can move, procedures for turning, hit and miss rules for shooting
  - Prioritize feature ideas, those at top are essential
  - Work through list
    - e.g. Moving and shooting crucial, require scoring and hit point attributes



# Features and Rules

- Features
  - Attributes that make game richer
    - e.g. "you can plant booby traps to surprise your opponent"
- Rules
  - Modifications to the game mechanics that affect how game functions
    - e.g. "You will lose points if you take a shortcut"
- New features need new rules



## The Steps in Prototyping: Step 3

- Develop formal details
  - Think about the formal elements of the game
  - What new elements are needed?
    - e.g. “ability to customize your vehicle before a race”?
  - Are the objectives interesting and achievable?
  - New rules or procedures?
  - **Beginners add in too much**
  - Test new rules individually
    - Remove optional ones



## The Steps in Prototyping: Step 4

- Refinement
  - Is it fun?
  - Revisit cool features
    - Rank them
    - Introduce individually into prototype



## The Steps in Prototyping: Step 5

- Game may feel unplayable, disjoint, slow etc.
  - Don't panic, debug
  - Strip away rules and reintroduce
- Now the game is playable and fun...
  - **Repeat**

