Lecture Week 3

Design Elements

the gamedesigninitiative at cornell university

- Verbs that describe what the player can do
 - **Walk**
 - & Run
 - **Y** Jump
 - **Shoot**
- Opes not need to be attached to an avatar
 - **Y** Build
 - **Y** Swap
 - **X** Rotate

Verbs that describe what the player can **do**

Walk (left or right) (walk, but faster!)

X Run (up; jump/run for left or right)

Y Jump (left or right)

Shoot

Opes not need to be attached to an avatar

Y Build

Y Swap

X Rotate

Verbs that describe what the player can do

Walk (left or right) (walk, but faster!)

Run (up; jump/run for left or right)

Y Jump (left or right)

Shoot

Action Platformer

Opes not need to be attached to an avatar

Y Build

Y Swap

K Rotate

Verbs that describe what the player can do

Walk (left or right) (walk, but faster!)

Run (up; jump/run for left or right)

Y Jump (left or right)

Shoot

Action Platformer

Opes not need to be attached to an avatar

X Build

Swap

X Rotate

Designing Actions

- **Y** Starts with brainstorming the verbs
 - **Y** Define the types of verbs
 - Y Define the scope of the verbs
- **Y** Design Goals
 - Through verbs to avoid being too simple
 - But not so much to be confusing (verb bloat)
 - Ye Do the verbs *directly* achieve the goal?
- **Y** Each verb maps to a single **input**

Primary Actions





- Y How do verbs, goals relate?
 - Y Imagine there is no challenges
 - What verbs *must* you have?
- **Example**: Platformers
 - **Goal**: reach exit location
 - Y Only need movement verbs
 - Killing enemies is *optional*
 - **Y** Other actions are *secondary*
- **Design Goal**: Primary focus
 - Secondary verbs lead to bloat
 - **X** Add features with interactions

Interactions

- Y Not a *direct* action of player
 - Outcome of the game state
 - Can happen w/o controller
- **Example**: collisions
 - Accidental or player forced
 - Y May be bad (take damage)
 - May be good (gain power-up)
- **Other Examples**:
 - **8** Resource acquisition





the gamedesigninitiative at cornell university

Game Mechanics

Game mechanic

- Relationship between verbs and interactions
- Often call this relationship the "rules"
- **Gameplay** is manifestation of these rules

Example: Joust

- **Verbs**: Flap; go left or right
- **Y** Interaction: Collision with opponent
- **Rule**: If hit opponent, lower player dies

Verb Minimalism



- **Y** Keep verbs at a minimum
 - **Y** Mechanics are all interactions
 - Common in mobile, tablet
 - Y Due to lack of input modes
- **Example**: Sneak Beat Bandit
 - Has only one verb: move
 - Khythm game; move to beat
 - X All movement on rails
 - ∀If obstacle in way, turn

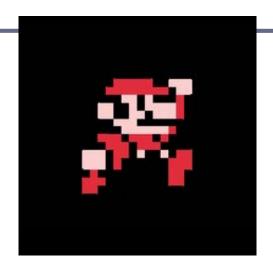
Avoid Verb Proxies

- **Proxy**: verb that activates another verb
 - **Y** "Use an item" (what does the item do?)
 - **Y** "Shoot" (what does the weapon do?)
- Make your verbs outcome oriented
 - Y Fire standard bullet (like shoot, but says what it shoots)
 - Y Fire freezing beam (what it does and how it is applied)
- **Y** Important questions to ask
 - Y Does it help me reach a goal?
 - **Y** Does it overcome a challenge?



Combining Actions

- Verbs can combine in interesting ways
- **Run** and **jump** in a platformer
- **Strafing fire in a shooter**
- Typically result of the interactions
 - Each verb interacts with environment in different way



Combining Actions

Running Jump

- Can move while in midair
 - Y Just horizontal movement
 - Y Not realistic; it is a game
 - Many platformer challenges assume this type of control
- Oifferent than a *long jump*
 - Y Less height than reg. jump
 - Y No control once in the air

Strafing Fire

- Based on "real life" property
 - Bullets travel in straight line
 - Y Movement changes origin
 - Walking side-side makes a spray (used in covering fire)
- But some features are gamy
 - Bullets slower than life
 - Character faster than life
 - **Creates interesting effects**

the gamedesigninitiative at cornell university

Challenges

Obstacles

- Y Prevent progress towards goal
- Have to be "overcome"

Opponents

- Y Players with their own goals
- May or may not need to be overcome

\(\) Dilemmas

- Can only perform one of several actions
- The "Correct" choice not immediately clear

Challenges: Limitations

- You cannot always perform an action
 - Shooting may require bullets Resource
 - Cannot (always) jump in mid air
- **Limitation**: requirement to perform action
 - Boolean test (like an if-then)
 - Checked at time of user input
- Only one limitation per verb
 - If more than one, split into more verbs

Challenges: Resources

- Game State: numeric and symbolic values that represent the game world at a specific moment
- **Numeric** quantities are often called **resources**
 - **Examples** (player): bullets, health points
 - **Examples** (external): monster spawns
- Symbolic values are "yes-or-no" quantities
 - Used for "lock-and-key" challenges
 - Y Typically create shallower gameplay



Putting It All Together

- Start with your vision
 - I want to _____
 - This creates setting and player goals
- Create a (partial) list of the following:
 - **Objectives**
 - **Actions**
 - **Y** Interactions
 - **Challenges**

Sketch player modes to show them in action



Questions?