



the
gamedesigninitiative
at cornell university

Rules and Mechanics

About This Lesson

- Material is from Unit 2 of *Rules of Play*
 - Available from library as **e-book**
 - Linked to from the lecture page
- Not required, but excellent resource
 - Important for the serious designer
 - And ignore the Amazon reviews...
- The “Bible of Game Mechanics”

What are Rules?

- Definition from *Rules of Play*:
 - Rules are *formal schemas*
- But what does this really mean?
- Is it different for digital games?

Challenge of Defining Rules

- They do not need to be *fixed*
 - **Example:** *Nomic* (simulates democratic voting)
 - But are changed in structured ways
- They can *ignored*
 - House-rules that add or remove rules
 - Rule relaxation (e.g. playing with a young child)
- They are not always *explicit*
 - **Example:** does *Battlefield* have rules on camping?

Implicit Rules

- We often consider these **social conventions**
 - If violate them, no one wants to play with you
 - Encapsulate being a “good sport”
- Implicit rules for Tic-Tac-Toe
 - Must move in a “reasonable” amount of time
 - If loss is inevitable, must move or forfeit
- These rules are generally made **ad-hoc**
 - Make them explicit only if there is a problem

Implicit Rules in Digital Games

- Often implemented as **terms of service**
 - Rules against using mods, bots
 - Rules against play-style (e.g. camping)
- Depend upon context, and can change
 - Ranked vs. unranked in network play
 - Official vs. private game server
- Exist because cannot specify everything
 - Goal is to prevent customer “churn”

How to Design Good Rules

- Player must have *meaningful choices*
 - Player must be able to make decisions
 - System must respond in significant way
- **Bad Rules:** Guess heads or tails to pick a winner
 - All you can do is guess the answer
 - Has no significant effect on the outcome
- **Bad Rules:** Move pieces on board with no interaction
 - Actions have no meaning since pieces don't interact
 - There are no victory conditions or even challenges

Informal versus Formal Rules

Informal

- Part of initial design process
 - Focuses on how it looks
 - Less concerned with code
- Many span multiple frames



Formal

- Part of implementation
 - Corresponds to code
 - Defined at the frame level
 - Interactions link multiple animation frames together
- Goal: match informal design
 - Is behavior correct?
 - Is behavior expected?

Understanding Game State

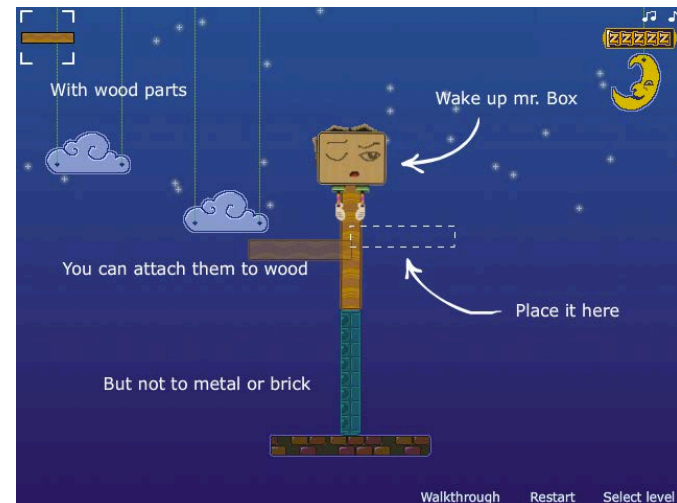
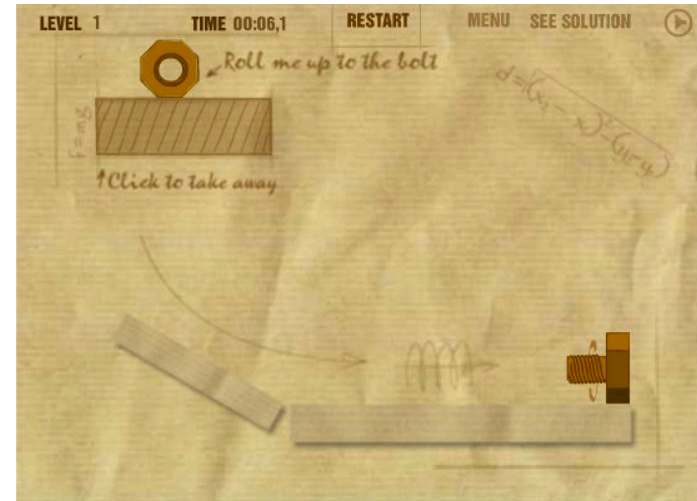
- Many game state values are **spatial**
 - Represent location of a game *entity*
 - Also physical values like velocity, acceleration
- Entities act as containers for non-spatial values
 - Values that never change: **attributes**
 - Values that can change: **resources**
- Attributes, resources can be global as well
 - Though most mechanics are at entity level...

Actions Affecting Spatial State

- Typically we what we would call *movement*
 - Present in all but the most abstract games
- But there are many ways to implement
 - **Direct** movement of avatar (e.g. WASD)
 - **Indirect** movement of avatar (e.g. pathfinding)
 - Alter the **environment** (e.g. removing platforms)
- Area of much potential *innovation*

Altering the Environment

- Found in “physics” games
 - No direct control of avatar
 - Can only remove/add/move obstacles in environment
 - Movement is “natural”
- **Example:** *Screw the Nut*
- Physics is a rule system
 - Interaction, not action
 - Takes one state to another
 - Also one that is complex to understand/model



Innovating Avatar Movement

- 2D games move on 2-axes
 - Classic: left-right/up-down
 - Unless top-down game, one of these axes is restricted
- Is jump the only option?
 - Launcher/trajectory verbs
 - (Limited) teleportation
- **Example:** *Knightmare Tower*
 - Launcher-style game
 - Vertical movement is boosts gained from killing enemies

Others?



Environment **AND** Avatar

- Possible to split the verbs
 - Some for avatar movement
 - Others for environment
- Found in “drawing” games
 - Draw missing platforms
 - Avatar walks on platforms
 - **Ex:** Max & Magic Marker
- Innovate by limiting avatar
 - Move on single axis
 - Combine with environment
 - **Example:** Swindler



“Deep Gameplay”

- Want many ways to overcome challenges
 - **Example:** kill enemy or sneak past
 - If just one way, gameplay is “shallow”
- Shallow challenges hurt replayability
 - “Twitch” challenges become boring fast
 - Cerebral challenges solved by the walkthrough
- All games should have a **strategic** element

Strategy

- **Definition:** an elaborate sequence of steps
 - Action is the culmination of all the steps
 - Changing steps or order changes action
- Still allows for puzzle gameplay
 - Allow some *flexibility* in these solution steps
 - **Example:** Multiple solutions to Rubik's Cube
 - **Example:** Time-rewind in *Braid*
- *Resources* are a common way to implement

Resources and Gameplay

- Resources are crucial to “combat” mechanics
 - Entities have resource values (e.g. *health*, *ammo*)
 - Expend resources to affect others (e.g. *attack*)
 - May change resources of that entity (e.g. *damage*)
- Three basic categories of resource combat
 - **Tug-Of-War**: entities take from each other
 - **Dot Eating**: entities race to gather *limited* resource
 - **Flower Picking**: race to gather *unlimited* resource

Resources and the Game Economy

- **Sources**: How a resource can increase
 - **Examples**: ammunition clips, health packs
- **Drains**: How a resource can decrease
 - **Examples**: firing weapon, player damage
- **Converters**: Changes one resource to another
 - **Example**: vendors, *Starcraft* barracks
- **Traders**: Exchange resources between entities
 - Mainly (but not always) in multiplayer games

Economic Challenges

- You can use resources to
 - Control player progression (hinder or advance)
 - Modify player abilities (limit or enhance)
 - Create a large possibility space (for replay value)
 - Create strategic gameplay
- Do not need a lot of resources
 - Not every game is a strategy game
 - But **almost all** games have some economy

Resources as Dilemma

- Players perform cost-benefit analyses
 - **Cost:** resource change not beneficial to player
 - **Benefit:** resource change beneficial to player
- **Example:** Survival Horror
 - Use ammo to shoot zombie (**Cost:** ammo)
 - Use knife to stab zombie (**Cost:** health)
 - Benefit the same in each case
- Players act with least cost for benefit

Resources and Monetization

- Most resources are gathered in-game
- But some games allow **external sources**
 - Get resources from a friend on Facebook
 - Pay for resources with a credit card
 - Known as resource *monetization*
- Free-to-play, pay-for-stuff
 - Modern business model for online games
 - But BIG pushback right now (loot crates)

Emergent Behavior

- **Coupled Interactions**
 - Two mechanics that can happen at once
 - **Verbs:** jump AND run in a platformer
 - **Resources:** warrior AND archer in an RTS
- **Context-dependent Interactions**
 - Mechanics combine to give new behavior
 - **Verbs:** jump and run is new form of movement
 - **Resources:** warriors form wall to cover archers

Emergent Behavior

- **Coupled Interactions**

Key Word

- Two mechanics that can happen at once
- **Verbs:** jump AND run in a platformer
- **Resources:** warrior AND archer in an RTS

- **Context-dependent Interactions**

- Mechanics combine to give new behavior
- **Verbs:** jump and run is new form of movement
- **Resources:** warriors form wall to cover archers

Emergent Behavior

- **Coupled Interactions**

Key Word

- Two mechanics that can happen at once

- **Verbs:** jump AND run in a platformer

- **Resources:** warriors

Advantage: game complexity grows **nonlinearly**

- Mechanics combine to give new behavior

- **Verbs:** jump and run is new form of movement

- **Resources:** warriors form wall to cover archers

Examples of Emergent Actions

Running Jump

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

Strafing Fire

- Based on “real life” property
 - Bullets travel in straight line
 - 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

Examples of Emergent Actions

Interaction(?)

ump

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

Strafing Fire

Interaction

- Based on “real n property”
 - Bullets travel in straight line
 - 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

Emergent Actions



Is this an example?
Why or **why not**?

Common Spatial Interactions

Collisions

- Can effect *resources*
 - Player takes damage
 - Player gains power-up
 - Player-NPC transfer gold
- Can effect *spatial values*
 - Bounce off collision point
 - Swing from attached rope
 - Attraction to magnet/charge

Detection

- Examples:
 - Line-of-sight (w/ obstacles)
 - Spatial proximity
- Can have *direct* effects
 - Alarms in a stealth game
- Can have *indirect* effects
 - Tower defense targeting
 - Adjust NPC reactions

Resource-Spatial Interactions

Resource Affects Spatial

- Resources can unlock areas
 - Keys are a trivial resource
 - Also use resource thresholds
 - **Ex:** Collect all tokens to pass
- Resources affect difficulty
 - Adjust input device sensitivity
 - **Ex:** Deadeye meter in *RDR*
 - **Ex:** Jet packs to increase jump

Spatial Affects Resources

- Resources made by entities
 - Have a spatial location
 - **Ex:** Time to transfer resources
 - **Ex:** Sources be captured
- Resource values are entities
 - Take up physical volume
 - Need space to acquire
 - **Ex:** Inventory in *Deux Ex*

Resource-Spatial Interactions



Spatial Affects Resources

- Resources made by entities
 - Have a spatial location
 - **Ex:** Time to transfer resources
 - **Ex:** Sources be captured
- Resource values are entities
 - Take up physical volume
 - Need space to acquire
 - **Ex:** Inventory in *Deux Ex*



Coupling is not Enough

- Example of *trivial* coupling:
 - RTS with single unit type – warrior
 - Coupling can arise from multiple warriors
 - When attack, count number on each side
- Group of warriors **is** sum of its parts
 - Just make a single warrior stronger
 - Discover from *resource analysis*
- Emergent behavior must couple *nonlinearly*
 - If n base mechanics, more than $O(n)$ behaviors



Example: *Starcraft*

- Basic units can
 - Attack in sky and/or land
 - Defend in sky and/or land
 - How can these combine?
- Further complexity:
 - “Buff” friendly units
 - “Control” enemy units
 - How does this affect game?
- **Challenge:** What is minimal complexity for a good RTS?



Summary

- Rules are **formal systems** defining your game
 - Specify to change the game state over a single frame
 - Challenge is matching them to your informal design
- **Resources** create *strategic* gameplay
 - Resources define the game economy
 - Strategy is just players making economic choices
- **Interactions** facilitate *emergent behavior*
 - Coupled actions/interactions creating new features
 - Can provide deep, nonlinear complexity