the gamedesigninitiative at cornell university

Lecture 24

Level Design

What is Level Design?

- Understanding of **player capabilities**
  - Abilities, mechanics available to the player
  - Assumptions of current player skill level
- Layout of **game geography**
  - Location and relationship of challenges
  - Movement of dynamic features (e.g. NPCs)
- Layout of **player progression**
  - How the player should move through the game
  - How the player visualizes this progression

Aspects of Game Design

- Games as **Education**
  - Train player skill and understanding
  - Focuses primarily on *player capabilities*
- Games as **Exploration**
  - Focuses on the *game geography*
  - Typically involves heavy storyboarding
- Games as **Storytelling**
  - Focuses on *player progression*
  - Most challenging element of game design

Aspects of Game Design

- Games as **Education**
  - Train player skill and understanding
  - Focuses primarily on *player capabilities*
- Games as **Exploration**
  - Focuses on the *game geography*
  - Typically involves heavy storyboarding
- Games as **Storytelli**
  - Focus
  - Most ment of game design

For a later lecture

Aspects of Game Design

- **Games as Education**
  - Train player skill and understanding
  - Focuses primarily on player capabilities
- Games as **Exploration**
  - Focuses on the game geography
  - Typically involves heavy storyboarding
- Games as **Storytelling**
  - Focuses on player progression
  - Most challenging element of game design

Learning How to Play

- Mechanics are (often) new and unfamiliar
  - Players have to learn how to interact with them
  - **Aside**: why innovation is not always popular
- Players could learn by reading the *manual*
  - This is boring! Let me play already
- **Tutorial levels** allow the player to…
  - Get started playing immediately
  - Learn the mechanics while playing

Classic Approach: Restrict the Player

- Start with your **gameplay specification**
  - Remove all but the barest mechanics
  - Remove verbs by disabling controls
  - Remove interactions by omitting "board elements"
- Levels add new mechanics back one at a time
  - **Example**: Platformer with a "no-jump" level
- Do not need to add a new mechanic each level
  - "Deep" mechanics allow many levels per mechanic
  - This can influence game geography (e.g. worlds)

Example: Starcraft Campaign

Explicit Restrictions

- Mechanics are unavailable for current level
  - Controls for actions are explicitly disabled
  - Interactions disabled, even if elements present
- **Motivation**: Prevents player confusion
  - Do not waste time on useless mechanics
  - Key in the casual and young audience
- **Examples**: Many AAA comercial games
  - *Starcraft* single-player campaign
  - *Portal* (integrated into story)

Implicit Restrictions

- Mechanics are always available, but not needed
  - Challenges designed for an explicit mechanic
  - Other mechanics may succeed, but they are harder
  - Level has hints to guide player to right mechanic
- **Motivation**: Allow replay in tutorial levels
  - Players go back and try optional approaches
  - Achievements are structured to encourage this
- **Example**: Many amateur Flash games
  - *My First Quantum Translocator*

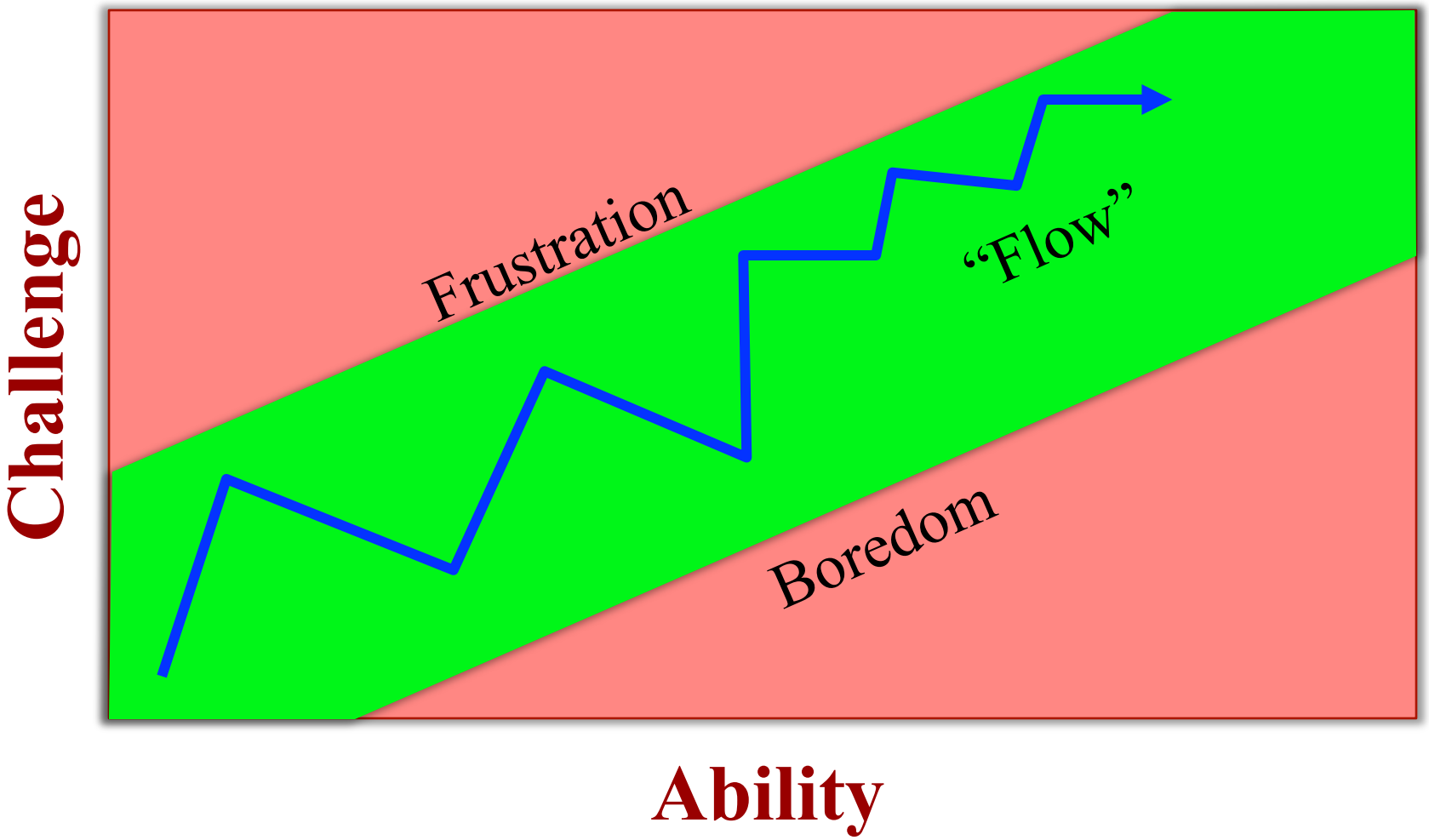
The Tyranny of Choice

- Too much choice can make us unhappy
  - We are often paralyzed by what to do
  - Studied by Myers & Lane; popularized by Barry Schwartz
- But games are about **meaningful choice**
  - Problem is when choices are too similar
  - Good choices must be *significantly* different
  - **Example**: Dagger adds +1 bonus to a stat of 102
- Players use rough heuristics for making choices
  - Pattern match current situation to determine action

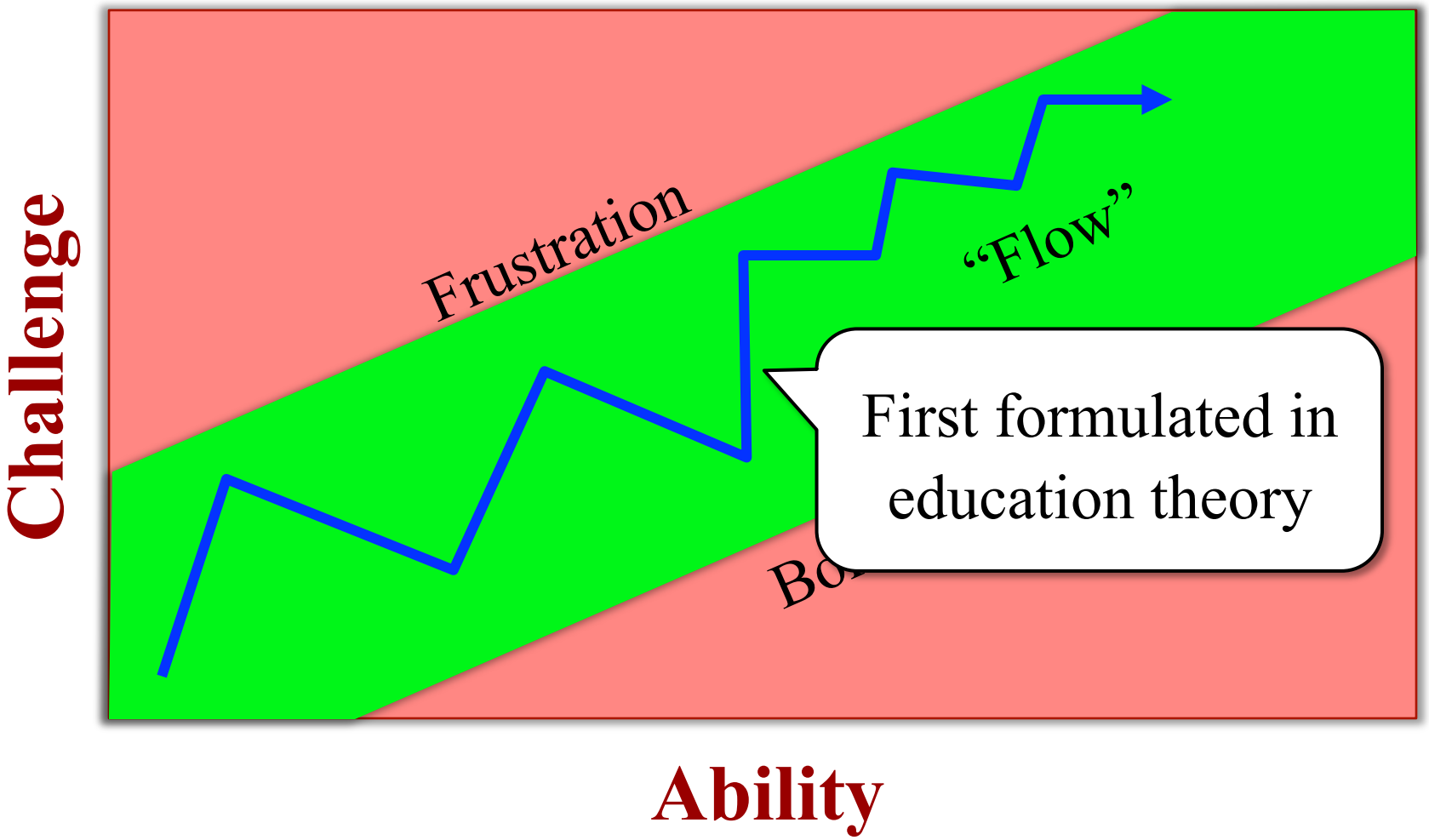
Leveling and Upgrades

- Natural way to restrict mechanics
  - Level gives access to new powers
  - Looting gives new items (verb proxy)
- Often act as a form of training
  - Demonstrate mastery in current mechanic
  - Mastery earns right to access new mechanic
- **Respecing** makes this learning process explicit
  - Player experiments with different mechanics
  - Finds right balance for current "game level"

Training and Flow



Training and Flow



Enabling Flow

- Challenging activity that requires skill
  - Could be physical, mental, or social
  - Impossible to someone without skill
  - Still uncertain to those with skill
- Clear **goals** and **feedback**
  - Player knows what must be done
  - Constant feedback on how achieved
  - Clear indication of failure or success

Steps to Designing a Tutorial Level

- Identify the **tutorial objectives**
  - What the player should be learning from game
  - Not necessarily the same as the game objective
  - In education, we call this a *learning outcome*
- Identify the **player assumptions**
  - What mechanics does the player understand?
  - How skilled is the player at the mechanics?
- **Storyboard** the player's progress

Aside: Puzzle Design is the Inverse

- Identify the **player assumptions**
  - What is the player used to doing?
  - How does player map patterns to choices?
- Create the **challenges** and **objectives**
  - Challenges should violate assumptions
  - Often an *interaction* player did not expect
  - Aided by *reinforcing assumptions* in early levels
- **Storyboard** the puzzle solution

Aspects of Game Design

- Games as **Education**
  - Train player skill and understanding
  - Focuses primarily on player capabilities
- **Games as Exploration**
  - Focuses on the game geography
  - Typically involves heavy storyboarding
- Games as **Storytelling**
  - Focuses on player progression
  - Most challenging element of game design

Players Want to Explore the World

- Exploring the **physical space**
  - What happens when I go here?
  - **Example**: Any western RPG
  - But does not require complex game world
- Exploring the **ludic space**
  - What happens when do this action?
  - Requires deep, complex interactions
  - **Example**: Buckets in Skyrim

Players Want to Explore the World

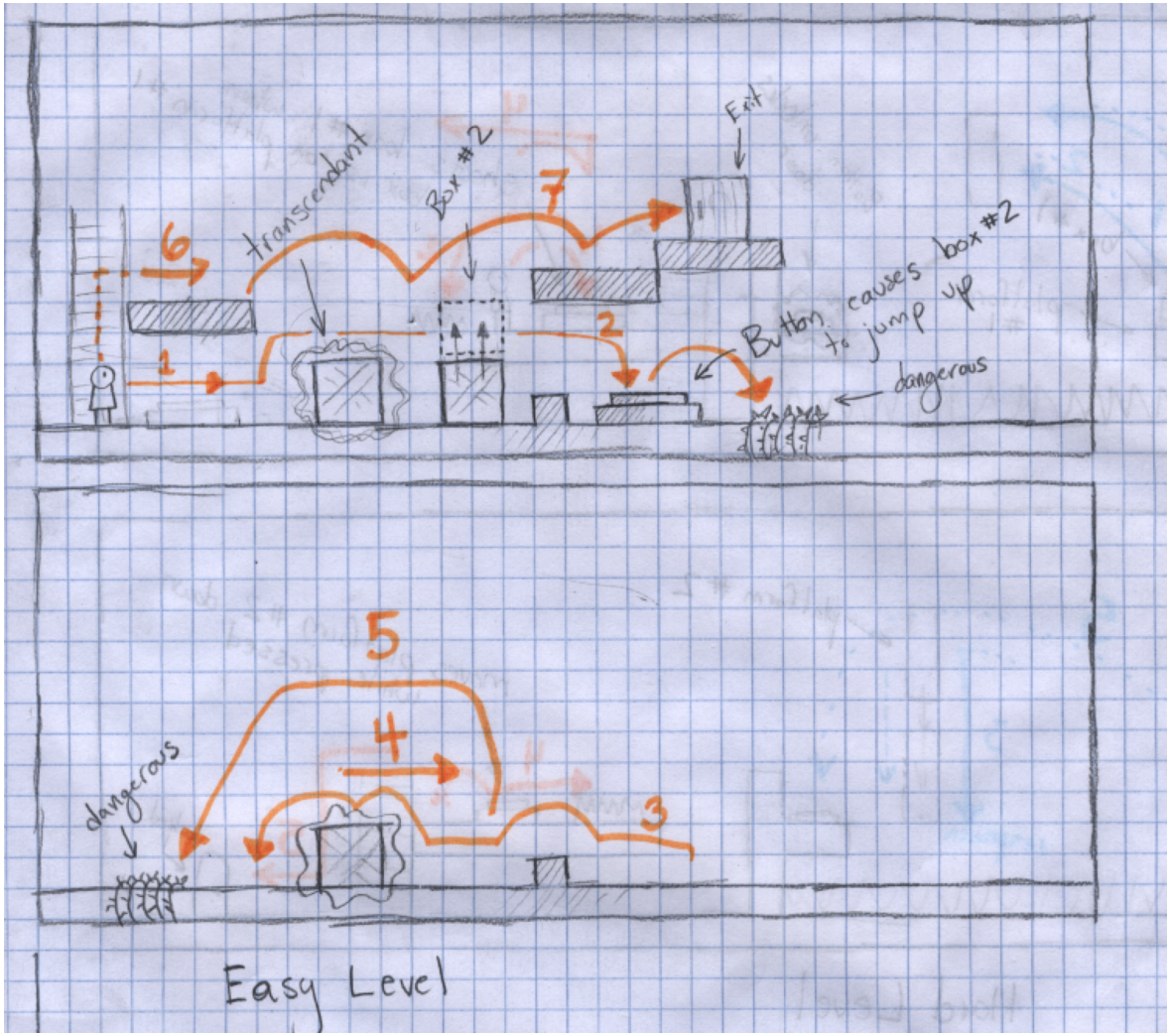
- Exploring the **physical space**
  - What happens when I go here?
  - **Example**: Any western RPG
  - But does not require complex game world
- Exploring the **ludic space**
  - What happ
  - E
  - **E**: Buckets in Skyrim

Essentially covered this already

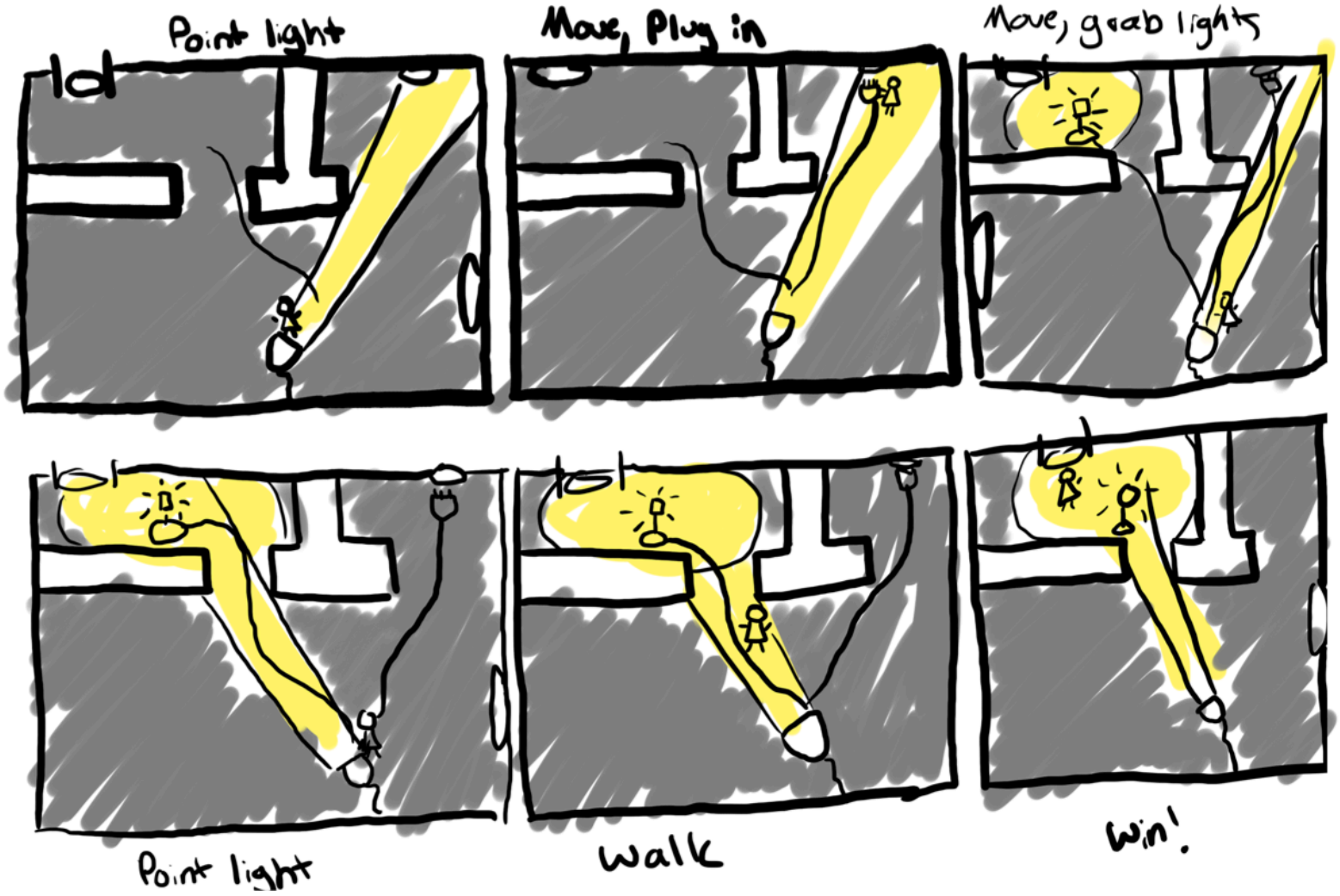
Storyboarding

- Diagrams player action throughout level
  - Different from film storyboarding
  - Currently a bunch of *informal practices*
- **Embodied Action**
  - Action that is tied to a character/avatar
  - Typically maps player movement in level
- **Disembodied Action**
  - Action corresponding to UI elements
  - **Example**: Buttons, menus

Embodied Action: Single Scene



Embodied Action: Multiple Scenes

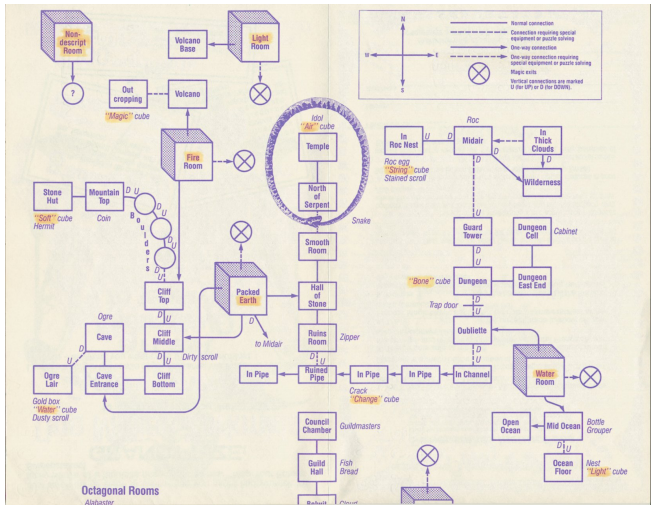


Disembodied Action: Cause and Effect

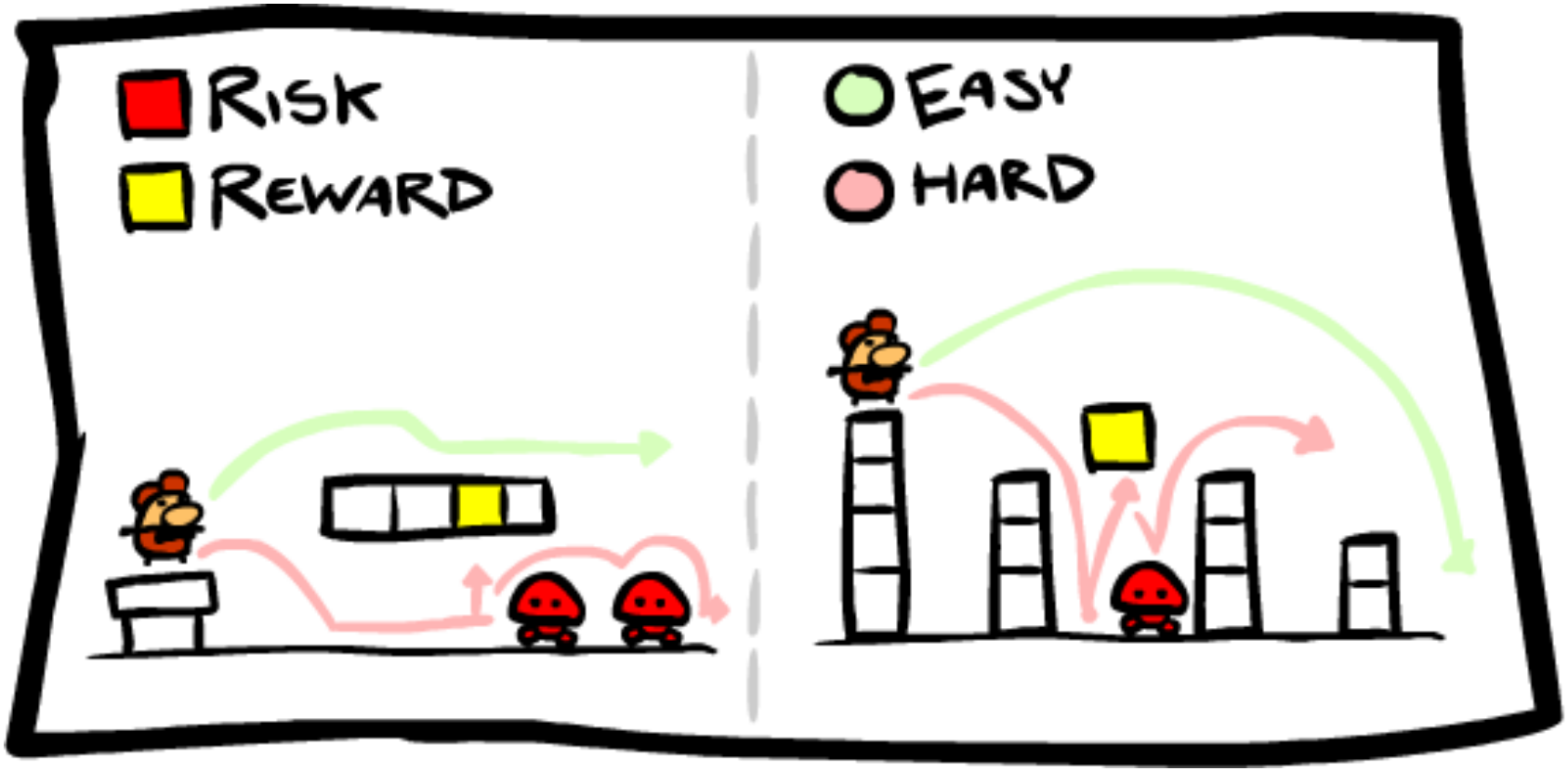
- **Draw the initial scene**
  - Could be the entire level
  - Zoomed in portion of screen
  - Must capture area that will be affected by the action
- **Indicate the action**
  - Draw mouse pointer
  - Indicate gamepad button
  - Annotate with a “tool tip”
- **Draw the action effect**
  - Change in initial scene

Game Geography

- Relations of game challenges
  - Multiple challenges in a level
  - Flow of level progression
- Easiest to design **discretely**
  - Well defined player paths
  - Some deviation allowed
  - Storyboard indicates paths
- Ensure **meaningful choice**
  - More than one path successful
  - Balance the risk vs. reward



Risk versus Reward



[Edmund McMillen, edmundm.com]

Risk vs. Reward in Ludic Space

- Player limits their actions
  - Avoids certain mechanics
- Rewarded for this restraint
  - Reward need not be ludic

Challenges with Geography

- World may be **too open**
  - Difficult to storyboard
  - Tyranny of Choice?
- World may be **dynamic**
  - Geography includes NPCs
  - They react to player actions
  - Again, how storyboard?
- **Discretization** is hard
  - Need to set *boundaries*
  - Must define *building blocks*

Boundaries

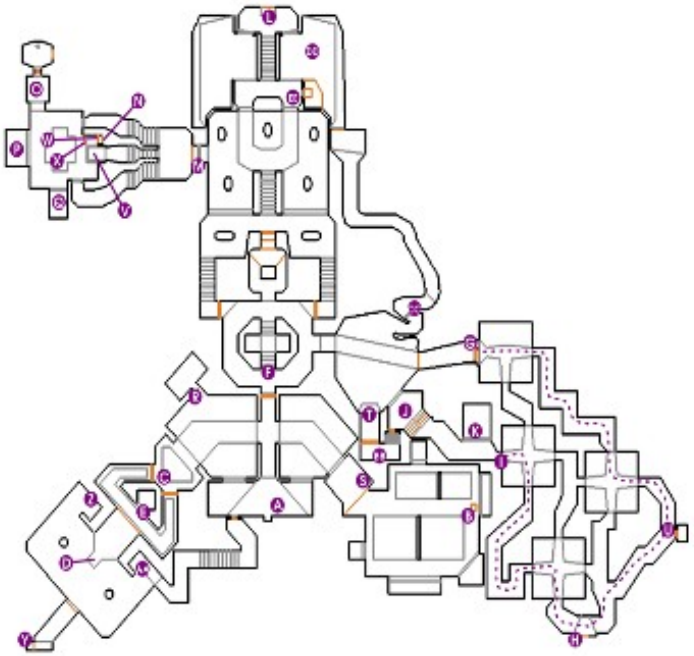
- **Explicit Boundaries**
  - Player is not allowed to go somewhere
  - **Example**: Platform layout in platformers
  - **Example**: Corridor layout in top-down or 3D
- **Soft Boundaries**
  - Player will face *unreasonable* challenges there
  - Player is at most "warned" about challenges
  - **Example**: Most open world RPGs

Level Design: Then and Now

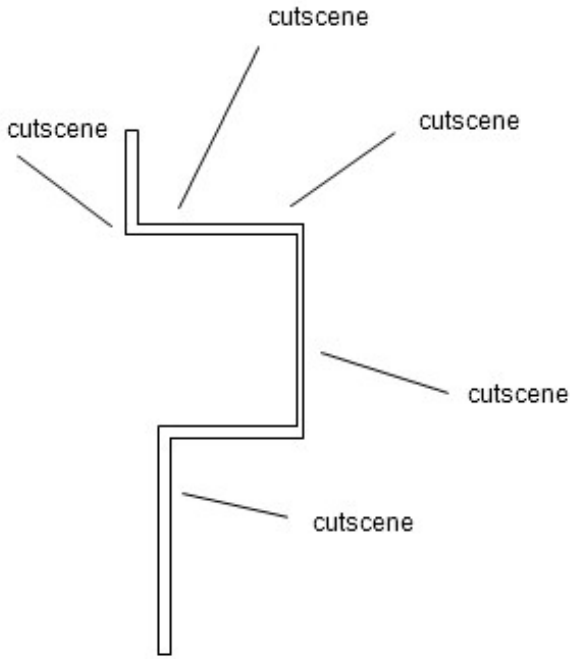
[Image attribution unknown]

FPS map design

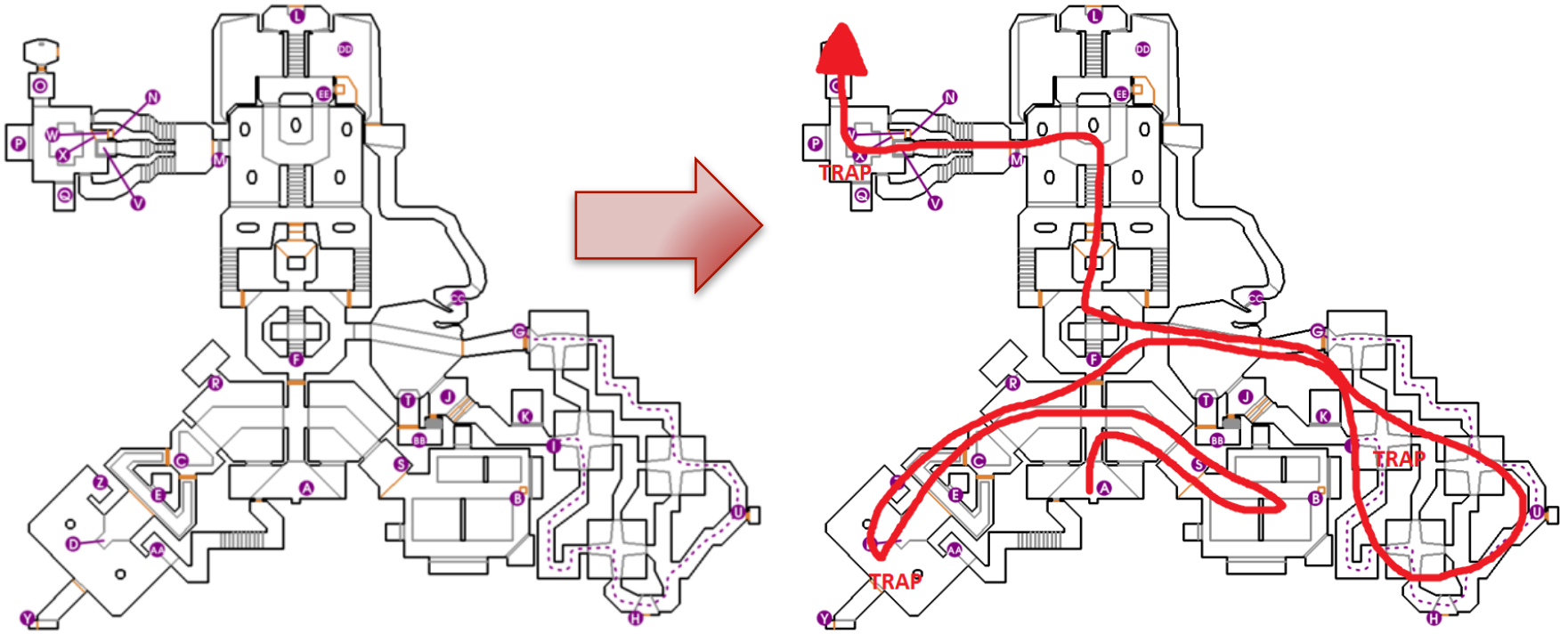
1993



2010



But Actually…

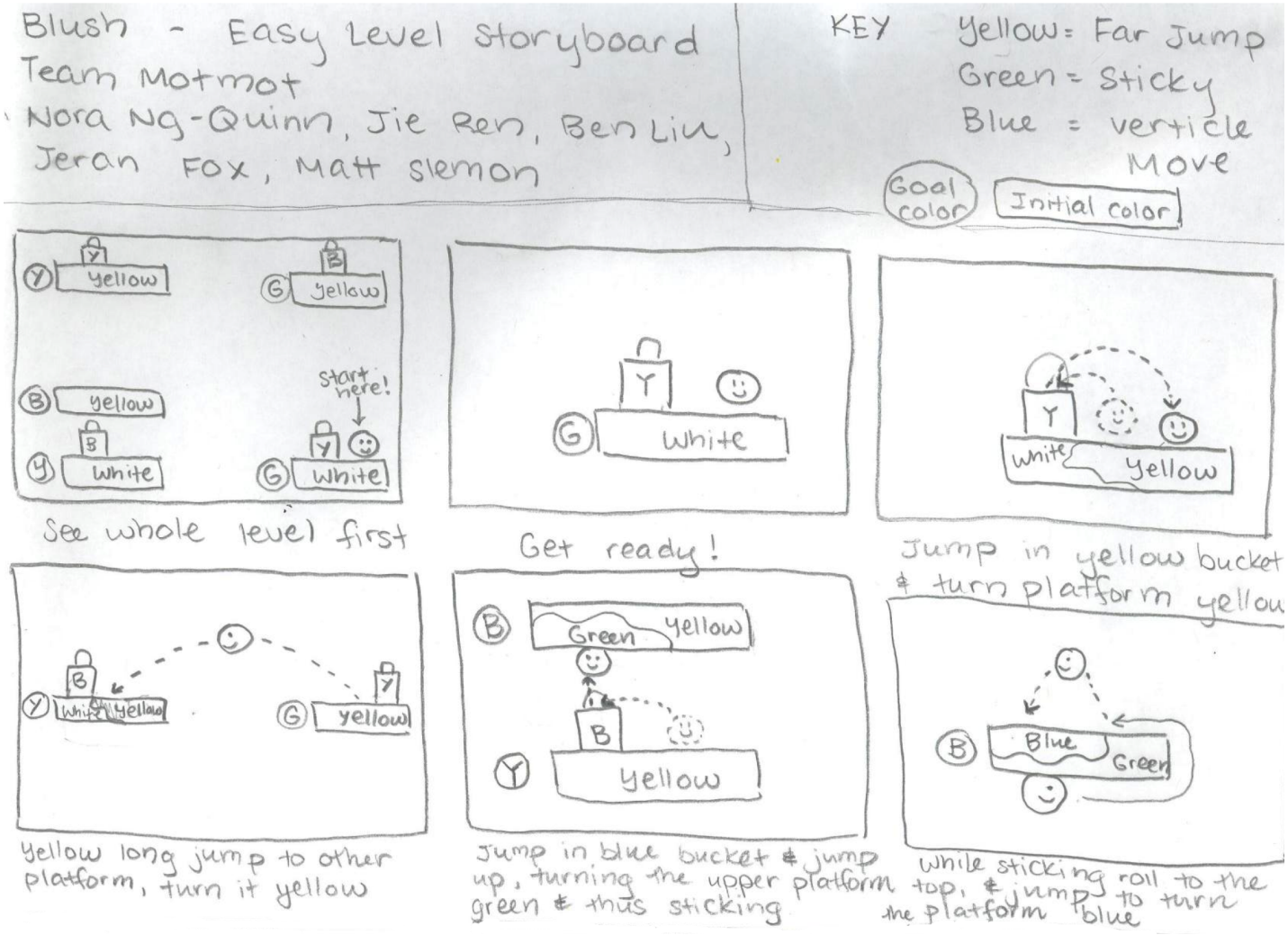


[refugeinaudacity.wordpress.com]

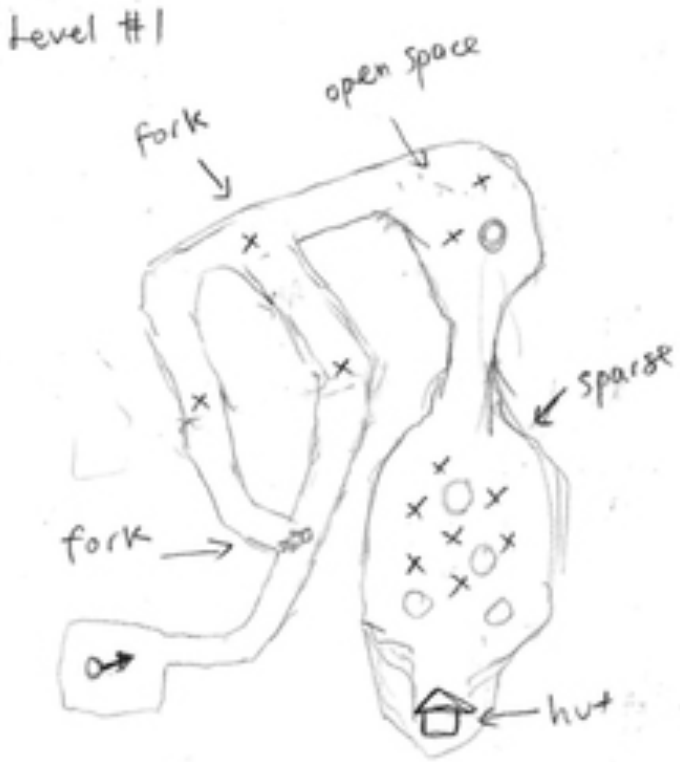
Building Blocks

- Design game as **individual challenges**
  - Single obstacle or mechanic usage
  - Single interaction with a single NPC
- Corresponds to a **scene** or **encounter**
  - Single frame in your storyboard
  - Much less that a whole level
- **Assemble** these together to get a level

Example: Blush



Example: Hollow Wood



Challenge Overlays

- Piecewise design creates a very linear feel
  - Challenge A followed by Challenge B followed by...
  - Player is explicitly aware of building blocks
- **Challenge overlays** allow for variations
  - Additional challenge added in same space
  - Makes the original challenge much more difficult
  - Player now has to react to them both
- **Example**: *cover busters* in shooters

Flanking and Cover Busters

Managing Challenge Overlays

- Should not be just another building block
  - Player sees it as "hard form of X”
- Player should have **control** over existence
  - Playstyle X is more likely to invoke an overlay
  - Overlay is associated with a reactive NPC
- **AI** is an integral feature to level design
  - Recognize when player is using a certain playstyle
  - Define NPC behavior to allow manipulation

Next Time…

- Games as **Education**
  - Train player skill and understanding
  - Focuses primarily on *player capabilities*
- Games as **Exploration**
  - Focuses on the *game geography*
  - Typically involves heavy storyboarding
- **Games as Storytelling**
  - Focuses on *player progression*
  - Most challenging element of game design