G5DGAM

Game Design – Progression

Progression

- Progression
  - Games become dull if the challenges never change.
  - Level design is about introducing new challenges to keep the player engrossed in the game.
  - New features for the player:
    - Abilities (e.g. attack moves, swimming, flying)
    - Equipment (e.g. weapons, armour, vehicles)
    - Characters (e.g. engineer, wizard, medic)
    - Buildings (e.g. garage, barracks, armoury)
  - New levels features to overcome:
    - Opponents (e.g. with new abilities, buildings or equipment)
    - Obstacles (e.g. traps, puzzles, terrains)
    - Environments (e.g. battlefields, racing tracks, climate)
  - Type of progression helps to determine a game's identity.

Features

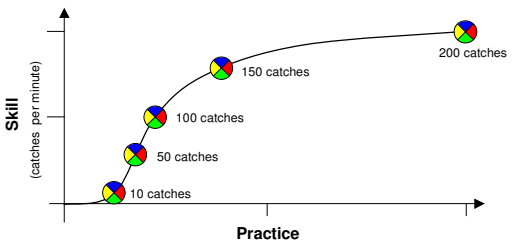
- Feature Considerations
  - Feasibility: prototyping, resources, timescales
  - Equivalent features: conning the player, 'arms race'
  - Reuse: 'one trick ponies', preferably:
  - Emergence: interaction, 'good' and 'bad' emergence, also between level designer and programmer.

Teaching and Learning

- The Player is NOT your opponent!
  - "The fundamental motivation for all game-playing is to learn" Chris Crawford, 1982
  - "[…] with games, learning is the drug […]when a game stops teaching us we feel bored" Raph Koster, 2005
  - If the player is the learner then the game designer is their their mentor!
  - Rise above the theatrical threat of failure and aim to help players to master your game – its more fun for everyone!

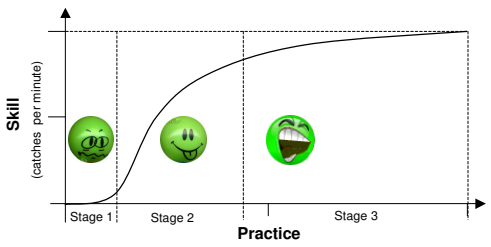
Learning Curves

- Learning Curves
  - Origins in manufacturing (experience against efficiency)
  - Learning to Juggle: catches per minute over time



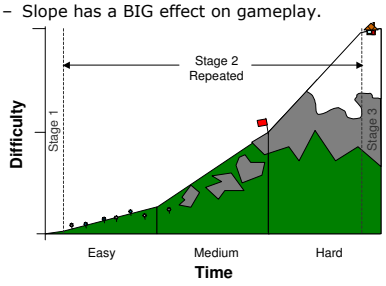
Learning Curves

- Learning Curves
  - Emotional states
  - Steep learning curve = easy?



Learning Curves

- Difficulty Curves
  - Not the same as a learning curve!
  - Slope has a BIG effect on gameplay.



Level Design

- A Heuristic For Level Progression (nothing more!)

1. Decide how many levels you want in your game.
2. Divide this total into three equal groups for EASY, MEDIUM and HARD levels.
3. Design each level and decide which group it belongs to:
   - All players should be able to complete EASY levels. Design these for players who have never played a game of the same genre before.
   - Most players should be able to complete MEDIUM levels. Design these for casual game-players of this genre.
   - Good players should be able to complete HARD levels. Design these for yourself and your friends who play these kinds of games.
4. If you end up with too many levels in one group then redesign some of them to make them harder or easier for another group.

Level Design

- A Heuristic For Level Progression (nothing more!)

5. Play all of the levels again yourself and arrange them in order of difficulty.
6. Test your levels on different players:
   - Friends or family members that dislike games of this genre should be able to complete the EASY levels without help (although bribery may be required!)
   - As the game's designer, you should be able to complete all of the MEDIUM levels without ever having to restart a level.
   - You friends who like games of this genre should eventually be able to complete all of the HARD levels without any help from you.
7. Tweak or reorder your levels according to the outcome of your tests.

Saving The Game

- Saving
  - Has a big effect on difficulty curves!
  - Beware of relying on it as a mechanic!
  - Use automatic saves or tokens instead

Koalabr8

- Applying It All..
  - New feature to encourage emergence: springs
  - 2 training levels: first with 1 koala then 2
  - 18 levels (6 easy, 6 medium, 6 hard)
    - Easy levels:
      - No time dependant features
      - Smaller playing area
    - Medium:
      - Time dependant features
      - Whole team synchronisation
    - Hard:
      - Cryptic and counter intuitive puzzles

Koalabr8

- Your Turn
  - Design and implement a new feature
  - Can you make one that encourages emergence?
  - Create a new level using the feature
  - Test it on each other
  - Review/redesign until you run out of time…