

CS 672: Spring 2010

Game Programming and Design

<http://www.rodvik.com/rodgames/>

Rapid prototyping
Abstracting visuals and interaction

Preface: A Word of Warning...

- The Game Design Kiss of Death (or, I hurt you because I love you)
- Would be great
- Reality
- So: try not to get too attached to your „great ideas“

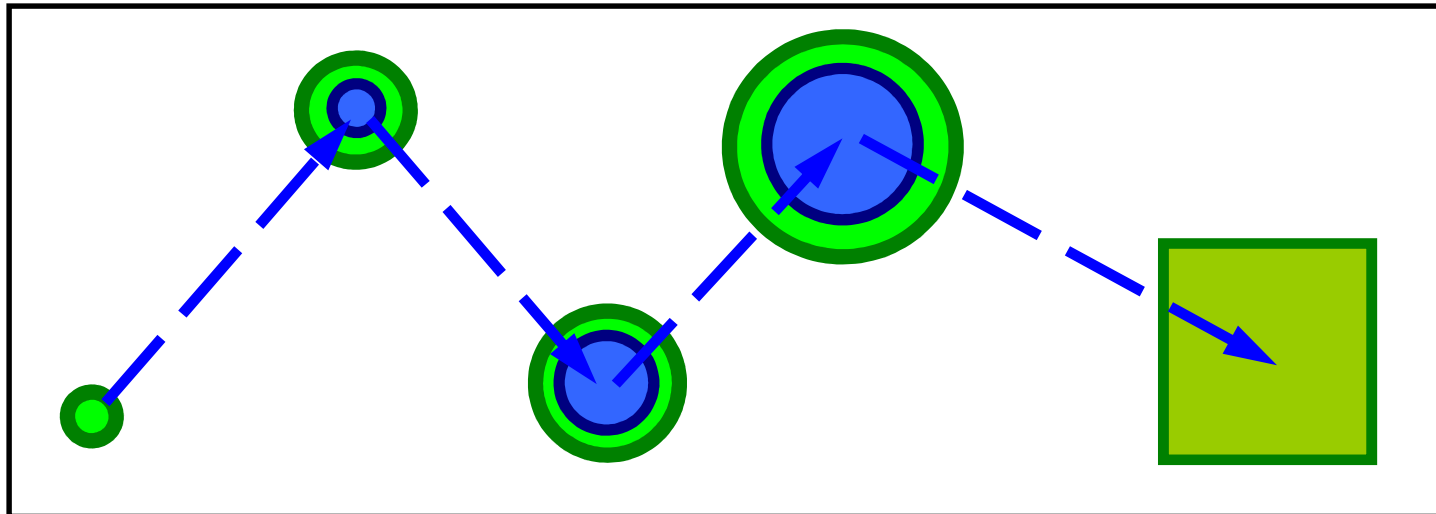


<http://www.2dboy.com>

[2DBoy 2007]

In the beginning...

- Start with rough prototypes
- Typical game situation
 - Need to eat ● and progress to ■ in order to evolve



In the beginning...

- A real-world example: flOw



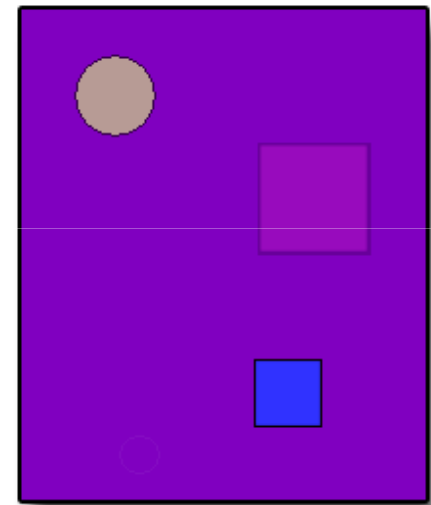
- Immersive ambient sound and visuals
- Easy to pick up and play
- Beautiful

In the beginning...

- Think in 2D for initial game designs
 - You can always move to 3D later
 - But first generate interesting and fun 2D object-to-object interaction and rules
- Check out other stuff and get inspired
 - This is not limited to games
 - Nature, physical/biological phenomena as well as personal issues often serve as good starting points
- Play (your) games a lot!

Game Design and Rapid Prototyping

Rapid Prototyping
is
Game Design



<http://www.rodvik.com/rodgames/>

But: game design is not necessarily
rapid prototyping

Prototype vs. Design Doc.

- Design Document
 - Very cheap
 - Static
 - Boring
 - Faith in developers
 - Tedious to parse
 - Tedious to create
- Prototype
 - Cheap
 - Interactive
 - Sexy
 - Science
 - Can be very complex
 - Can (and should) be thrown away and redone

[Gingold 2006]

Rapid Prototyping

- What is Rapid Prototyping ?
 - Fast, dirty demos of gameplay mechanics
 - Easy to implement, test, iterate and throw away
 - Have an idea, ask questions, build prototype [Gingold 2006]
 - Talk, build, talk again, build more... etc., etc.
 - Ask questions
 - Can moving a character through blobby obstacles be fun?
 - Surprising one self with new ideas
 - Speed/number of iterations correlates with good design
 - Works best if technology issues are **not** in the way


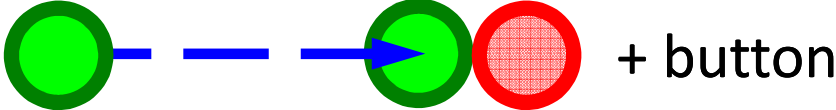
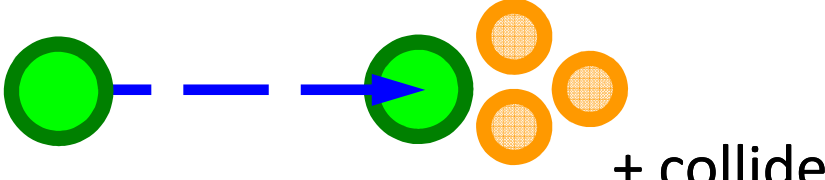
Rapid Prototyping

- What is Rapid Prototyping ?
- In a nutshell:

ABSTRACTION

Interaction Abstraction

- Abstraction examples of complex tasks

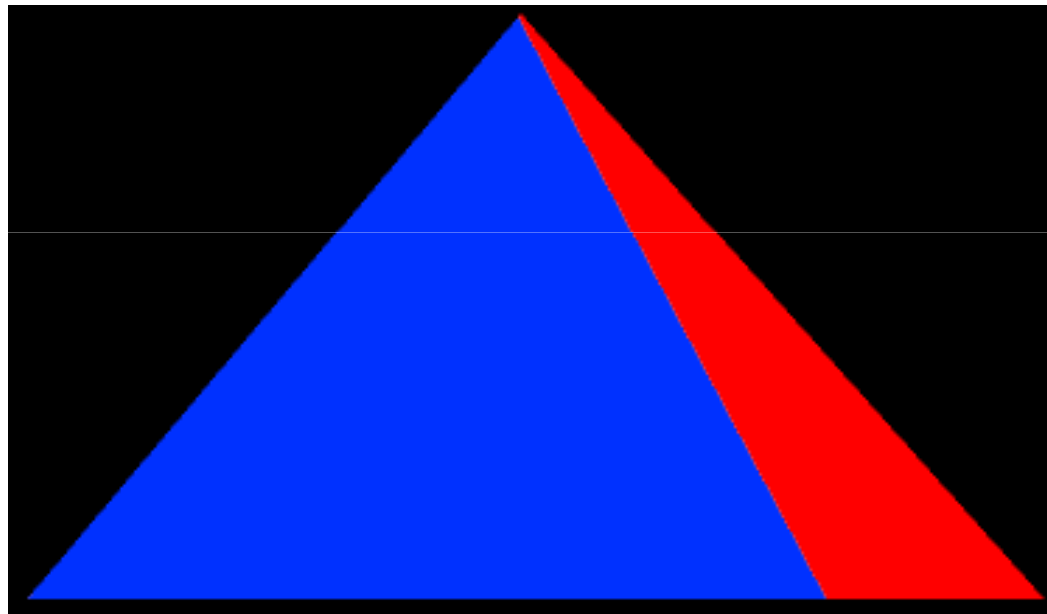
- Motion 
- Manipulation  + button
- Gathering  + collide

- Player should learn skills as the game progresses
(Many Nintendo games master this)

- Example: Metroid series. Use skill after acquisition
- Example: Zelda series. (Re)learn set of skills for final battle

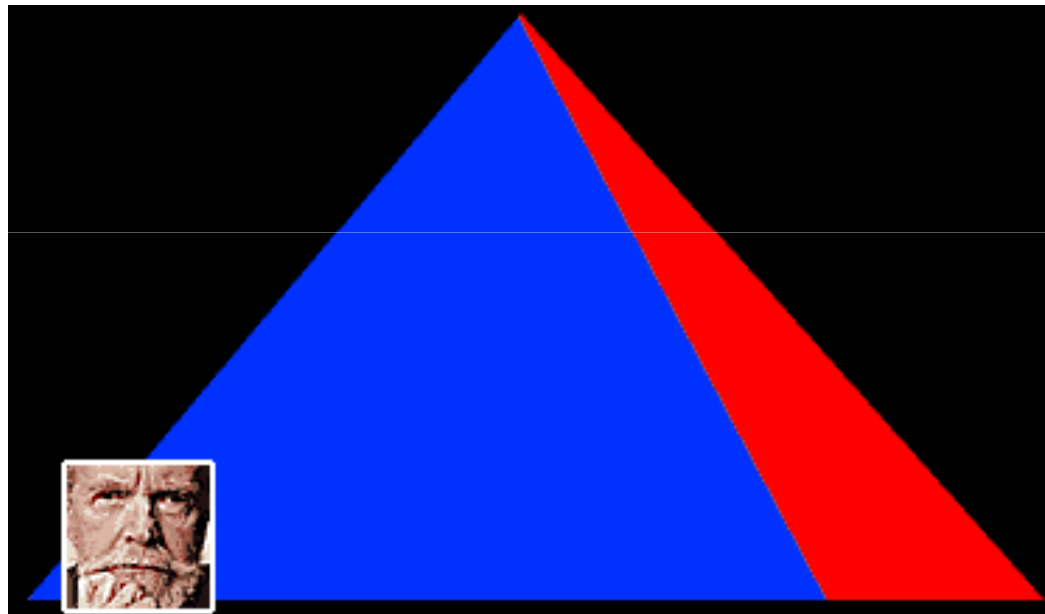
Visual Abstraction

- A map of visual iconography



Visual Abstraction

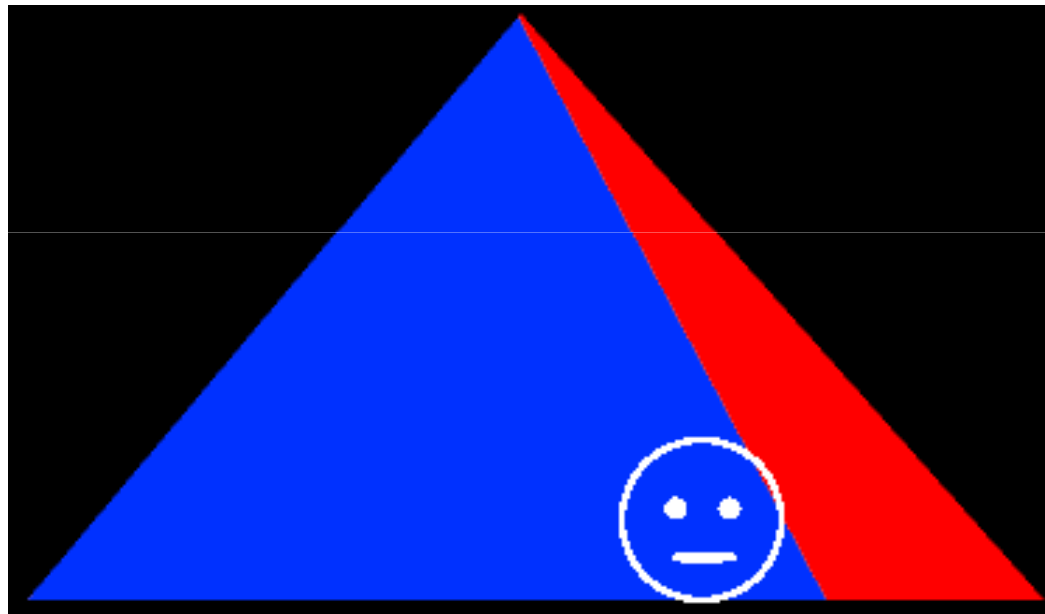
- A map of visual iconography



- Lower left: visual resemblance (e.g. photography)

Visual Abstraction

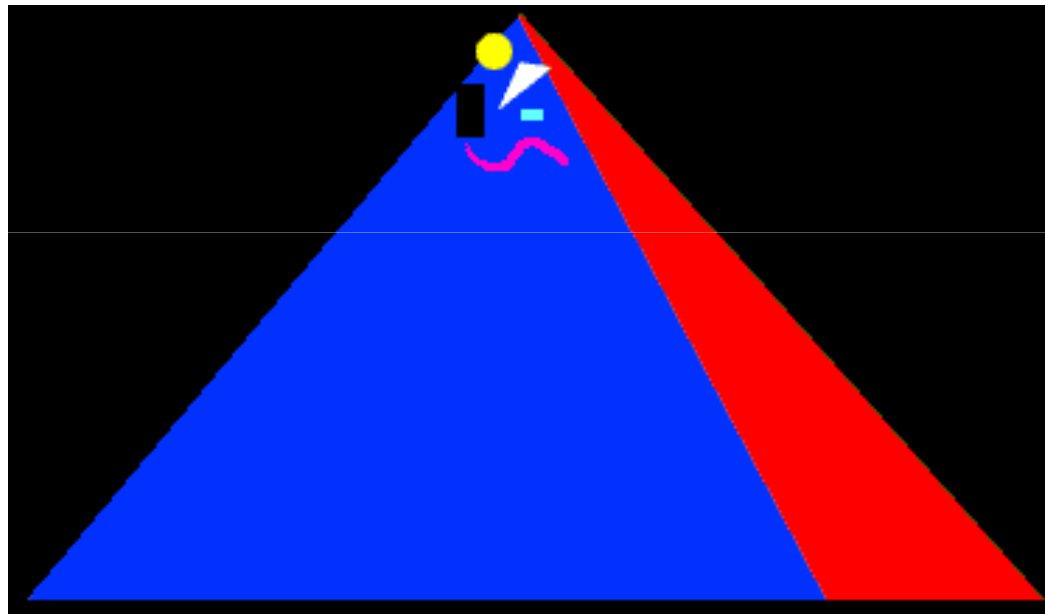
- A map of visual iconography



- Lower right: iconic abstraction (e.g. cartooning)

Visual Abstraction

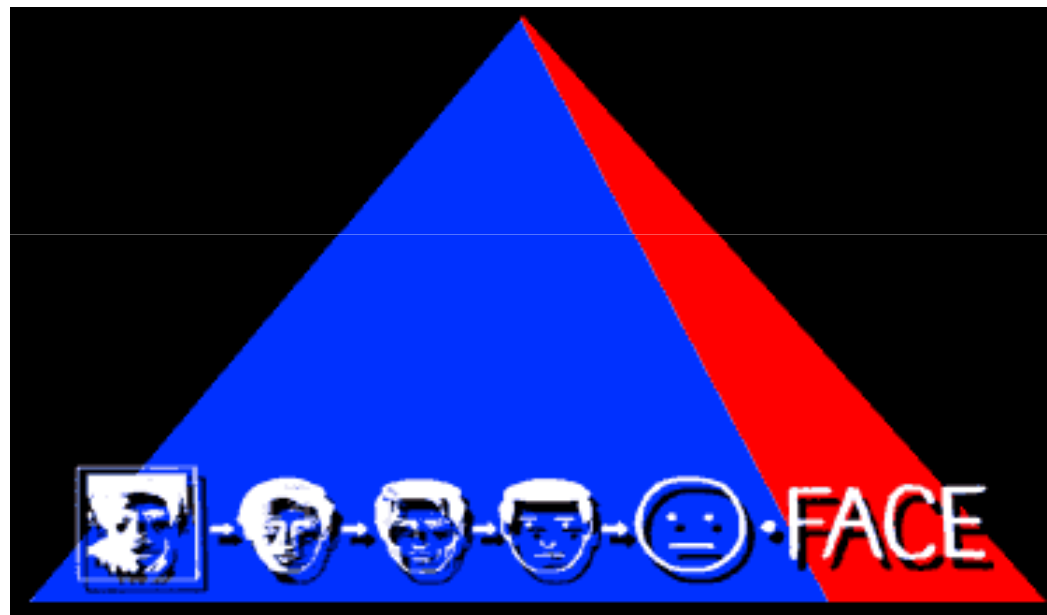
- A map of visual iconography



- Top: picture plane („pure“ abstraction)

Visual Abstraction

- A map of visual iconography

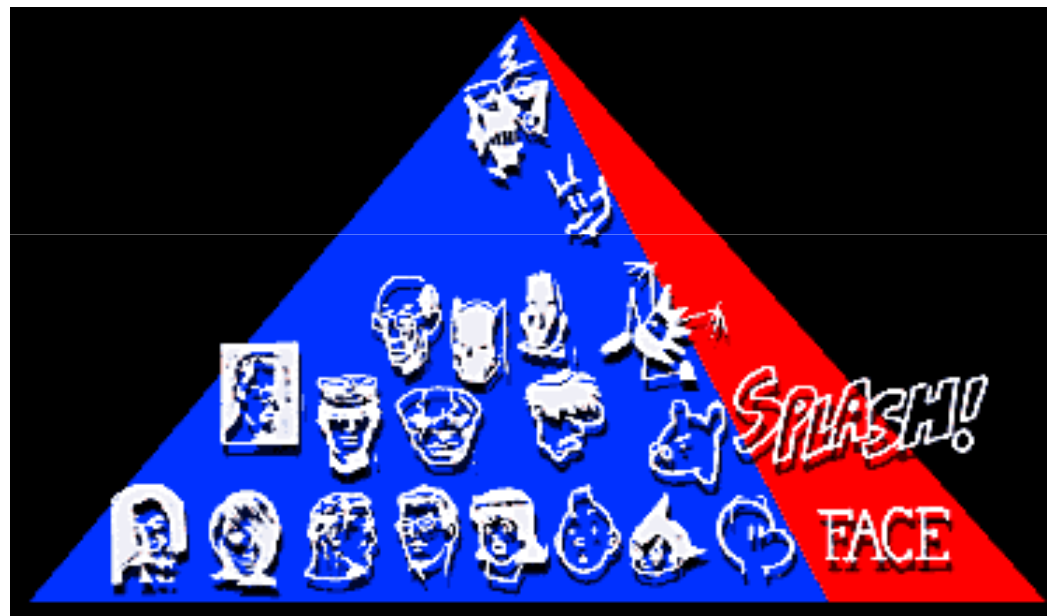


- Far right: from realism to cartoons... words as the next logical step

© Scott McCloud, *Understanding Comics*

Visual Abstraction

- A map of visual iconography

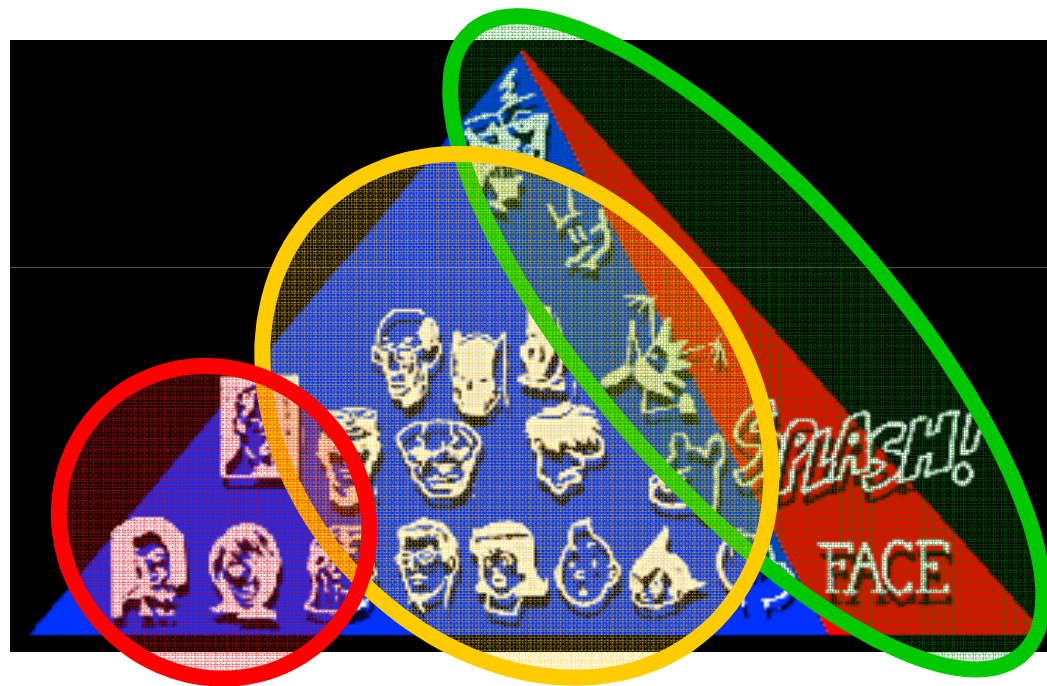


- Interesting tool for thinking about comics and games as art

© Scott McCloud, *Understanding Comics*

The Game Design Context

- Areas of game design iconography



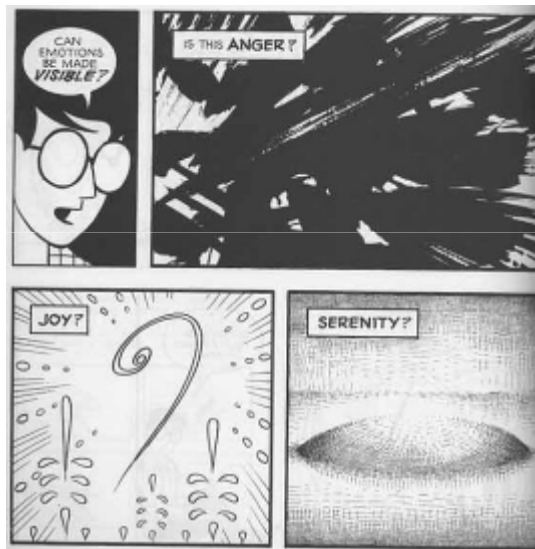
Potential
Uncanny Valley

Final Game

Prototype

The Game Design Context

- Iconography creates emotion



© Scott McCloud, *Understanding Comics* etc.

- **Leverage:** cultural knowledge, memory, nostalgia
- **Constructing Artificial Emotions: A Design Experiment**
http://www.gamasutra.com/view/feature/1992/constructing_artificial_emotions.php?print=1

Towards a Procedural Game Narrative

- A Thesis
 - A „game narrative“ can be expressed and conveyed through the **abstraction** of both **interaction** and **visuals**, and need not necessarily rely on explicit storytelling
- Further reinforcement of this thesis
 - Sound and music, for example Everyday Shooter (video)
 - Discernable and integrated actions and outcomes
 - Cultural and personal experience
- The Bottom Line:
Create **emotions** that drive player **motivation**