Killer Engine for Remixing Games

Jesse Himmelstein
CRI-Paris
New game engine, with different goals

Made for remixing games and mashups

Electronics-inspired visual programming

Excellent debugging and live coding tools

All done in the browser
Interdisciplinary science and engineering
Education technology

CTO @ cri-paris.org

- Knownodes
- Coldea
- SYNBI04ALL
  www.synbio4all.org
Scientific and Educational Games

Archimedes' Escape

Leap
Weizmann Institute

A SYNTHETIC BIOLOGY VIDEO GAME
ERO.COLI
What I Do

- Citizen Science Games

FoldIt
Univ of Washington

Fraxinus
The Sainsbury Laboratory
A scientist should just be able to take part of one game, mix it with another one, and then modify it to make a new one.

Why?

That’s impossible.
Why?

- We have:
  - Functions
  - Classes
  - Metaclasses
  - Entity Component Systems
  - Blocks
  - Unit tests
  - Distributed Version Control
  - End to end tests
  - Virtual machines
  - A million programming languages
  - A billion game engines
  - A kabillion libraries
Simplicity

- Interleaving -> incomprehension

- Lessons
  - Avoid tangling
  - Leave data alone
  - Sane abstractions for state and time

Rich Hickey
"Simple made Easy"
Inspiration

- Breadboard
  - Connect anything
  - Measure anywhere
  - Replace everything
Visual Programming

- Been done before...

Data flow

flowhub
Unreal Kismet
Putting order

1. Memory -> Chips
2. Chips compute
3. Chips -> Memory
1. IO & Memory -> Chips
2. Chips compute
3. Chips -> IO & Memory
Collaboration

- Memory
  speed
  position

- Move
  speed
  position

- IO
  mouse
  keyboard
  graphics

- Draw
  position
  shape
Chips

Emitter

Splitter

Processor

Switch
Switches

Switch

Chip

Chip

Chip
Cool debugging tools

- “What would have happened”
- “Mute” chips
- Find chips from memory or IO
- Test each chip in isolation

Brett Victor
“Inventing on Principle”
Sandboxing

- User code is dangerous
- Contain within iframe with sandbox attribute
- Messages replace function calls

Parent page - Editor

Sandbox - Game execution

iframe
sandbox="allow-scripts"
Patching

- Two chips modifying the same array/object?
- Like version control
  1. Create patches from a common ancestor
  2. Merge patches
- Some merges are hard
  - LCS (Longest Common Subsequence)
  - Operational transforms
Track *all* past state

Cloning data takes space!

Better solutions
- Store mostly patches
- Persistant data structures
  - Applies to lists, trees, hash maps, etc.

List A
[1, 2, 3, 4]

List B
[5, 2, 3, 4]
Treat IO as a buffer instead of a set of methods

Input
- Mouse (getMousePos() -> mouse.position)
- Keyboard (isKeyDown() -> keyboard.keysDown)

Output
- Canvas
  
```javascript
ctx.fillStyle = "black";
ctx.fillRect(0, 0, 100, 100);
```

```javascript
canvas.shapes = [{
  fillStyle: "black",
  position: [0, 0],
  size: [100, 100]
}];
```
Buffer IO

- Input/Output
  - HTML using Rivets.js
    - Template + values
    - 2-way data binding
  - HTTP
    - Data replaces events and callbacks!
Almost done...
We need you!

- New version landing Feb 14th
- Looking for new and different games
- Developers and designers always welcome!
  - Lots of new features to work on
Paris Game Club - Making, playing, learning

- Invited speakers
- Micro-game jams

Videos + agenda at

[gamelier.org](http://gamelier.org)
github.com/CyberCRI/RedWire
@RedWireIO
jesse.himmelstein@parisdescartes.fr
+JesseHimmelstein
@himmelattack