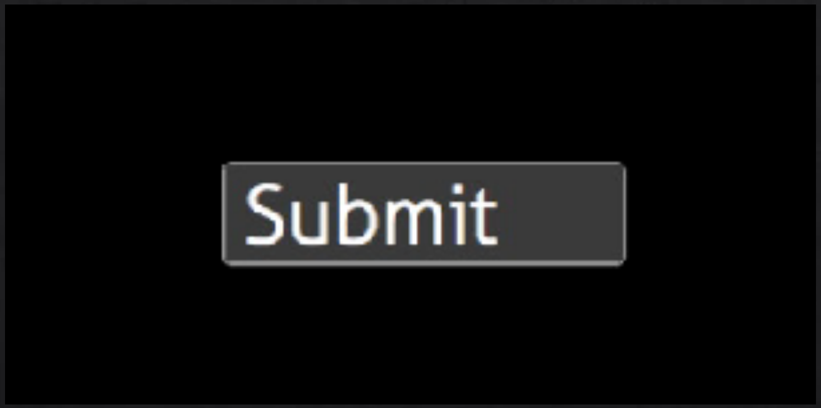
Expressing Interactivity with States and Constraints

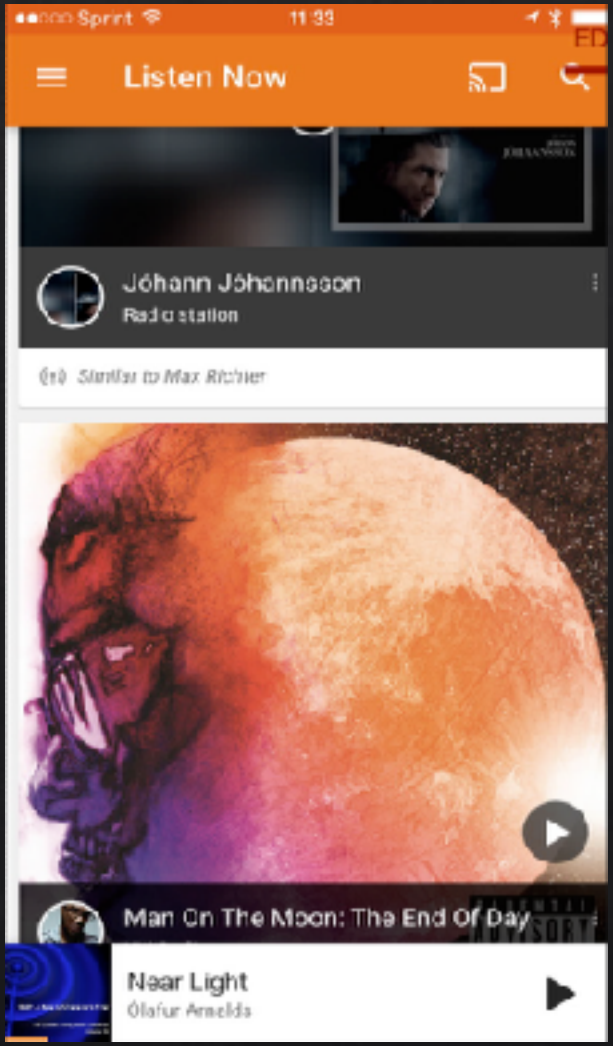
Steve Oney
University of Michigan
School of Information

- interfaces are defined by:
    - appearance (relatively easy)
    - behavior (relatively difficult)
        - reaction to user input (& other events)
        - controls usability & accessibility
        - sum of interactive behaviors

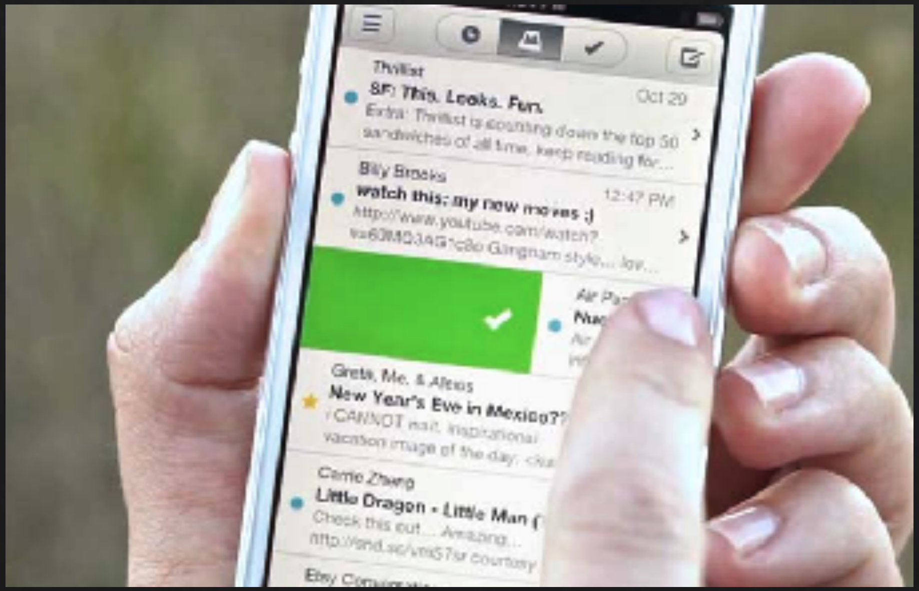
interactive behaviors



interactive behaviors



interactive behaviors

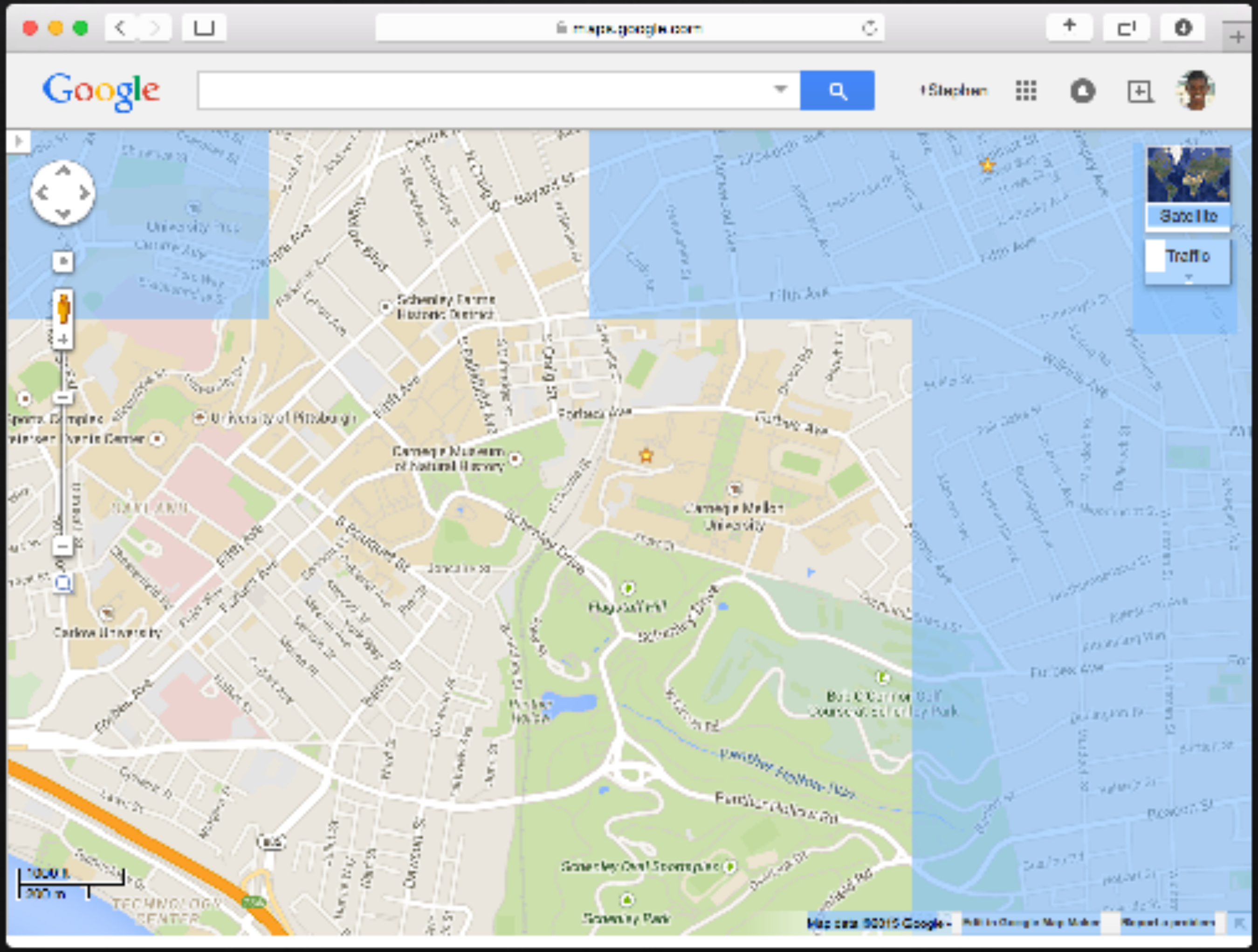


Mailbox

Tinder

a natural model for defining
interactive behaviors

interactive behaviors



Google Maps

gui programming

- reactive in nature
  - react to user events, timer events, etc.
- execution order (control flow) depends on external inputs

event-callback

- most common paradigm for defining interactive behaviors
    - specify what actions to take in reaction to events with callbacks

event-callback

on mouse move:

```
function mmove() {
    // ...
}
```

on mouse up:

```
function mup() {
    // ...
}
```

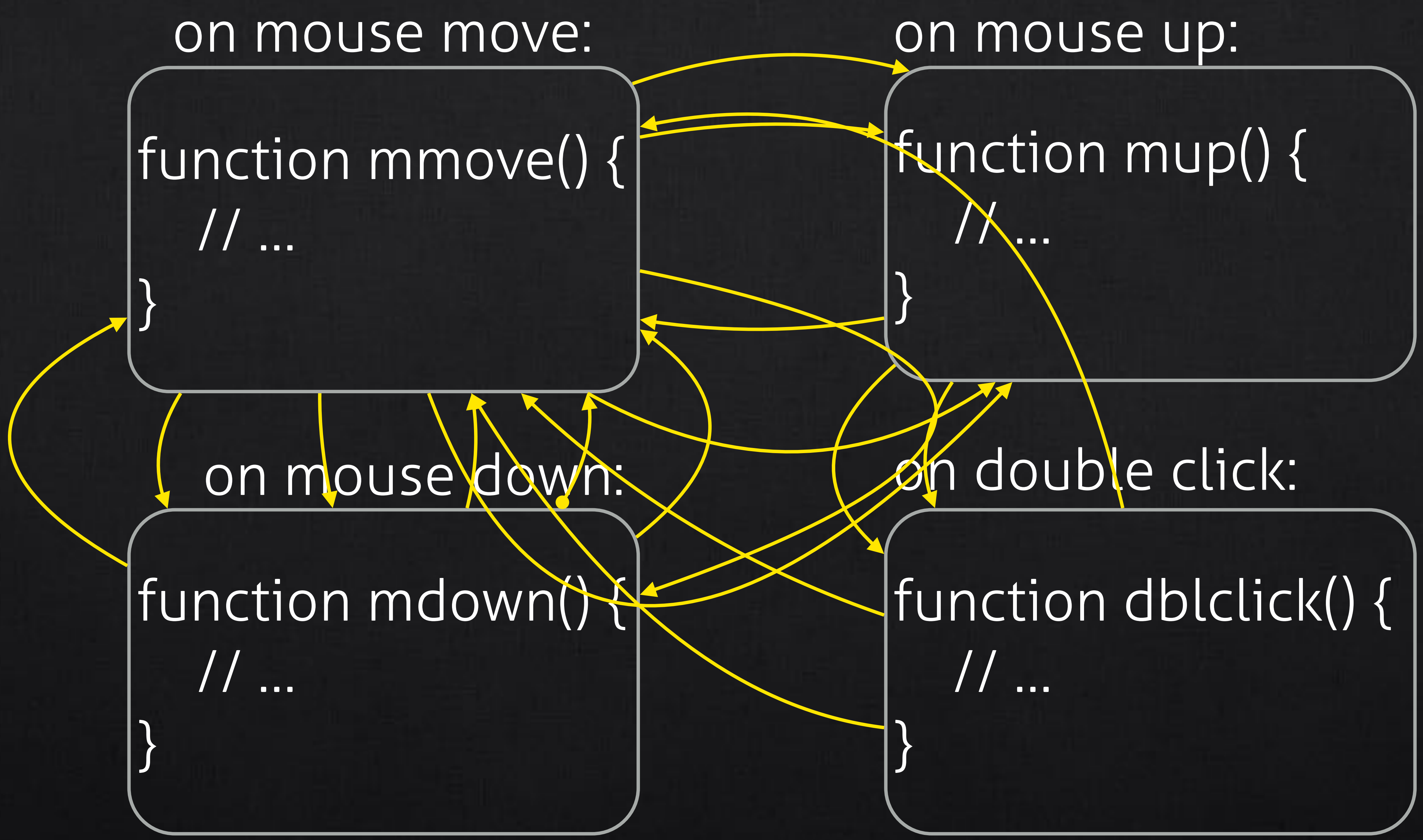
on mouse down:

```
function mdown() {
    // ...
}
```

on double click:

```
function dblclick() {
    // ...
}
```

event-callback



on mouse move:

```
function mmove() {
    // ...
}
```

on mouse up:

```
function mup() {
    // ...
}
```

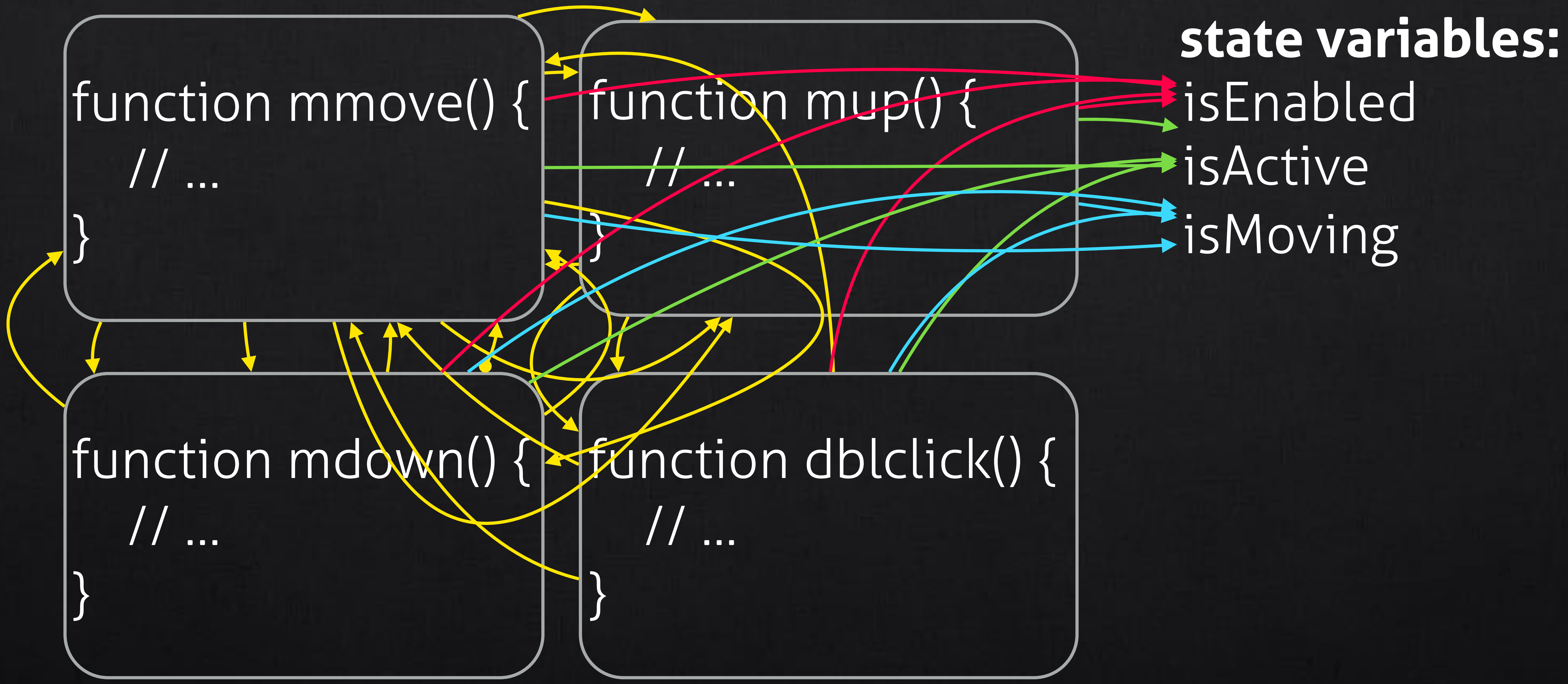
on mouse down:

```
function mdown() {
    // ...
}
```

on double click:

```
function dblclick() {
    // ...
}
```

event-callback



```
function mmove() {
    // ...
}
```

```
function mup() {
    // ...
}
```

```
function mdown() {
    // ...
}
```

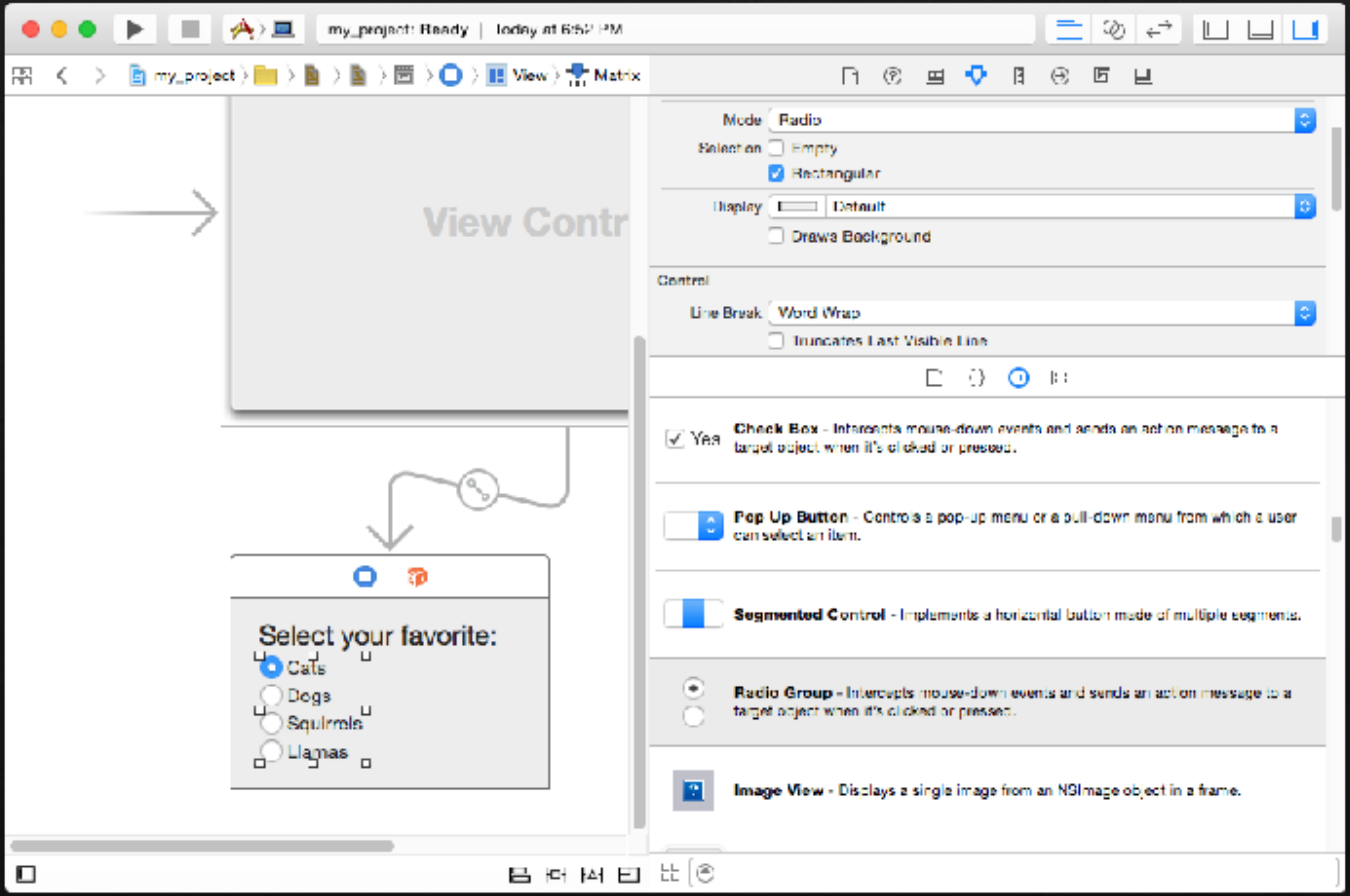
```
function dblclick() {
    // ...
}
```

state variables:
isEnabled
isActive
isMoving

event-callback

- leads to error-prone code
  - behaviors involve many callbacks
  - complex control flow for behaviors

[Letondal 2010, Myers 1991]



my_project: Ready | Today at 6:52 PM

my_project › View › Matrix

View Contr

Select your favorite:

- ● Cats
- ○ Dogs
- ○ Squirrels
- ○ Llamas

Mode: Radio

Selection: □ Empty

☑ Rectangular

Display: Default

□ Draws Background

Control

Line Break: Word Wrap

□ Truncates Last Visible Line

Check Box - Intercepts mouse-down events and sends an action message to a target object when it's clicked or pressed.

Pop Up Button - Controls a pop-up menu or a pull-down menu from which a user can select an item.

Segmented Control - Implements a horizontal button made of multiple segments.

Radio Group - Intercepts mouse-down events and sends an action message to a target object when it's clicked or pressed.

Image View - Displays a single image from an NSImage object in a frame.

Xcode

Select your favorite:

○ Cats
● Dogs
○ Squirrels
○ Llamas

Select your favorite:

○ Cats
● Dogs
○ Squirrels
○ Llamas

a natural model for defining
interactive behaviors

a natural model for defining
interactive behaviors

combines states and constraints

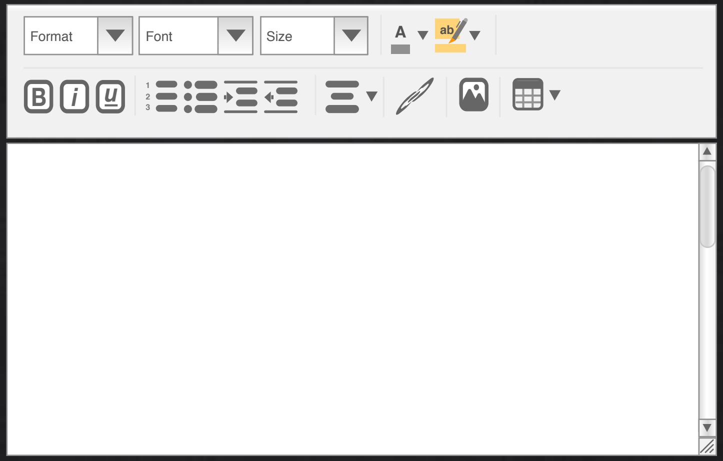
a natural model for defining interactive behaviors

principles:

- human-centric
- guided by:
    - non-programmers' descriptions of behaviors
    - workshops with interaction designers

[SY Park 2008, Ozenc 2010]

- “when the toolbar is docked, it is displayed above the workspace”
- “when the toolbar is being dragged, it follows the mouse”



- “when the toolbar is docked, it is displayed above the workspace”
- “when the toolbar is being dragged, it follows the mouse”

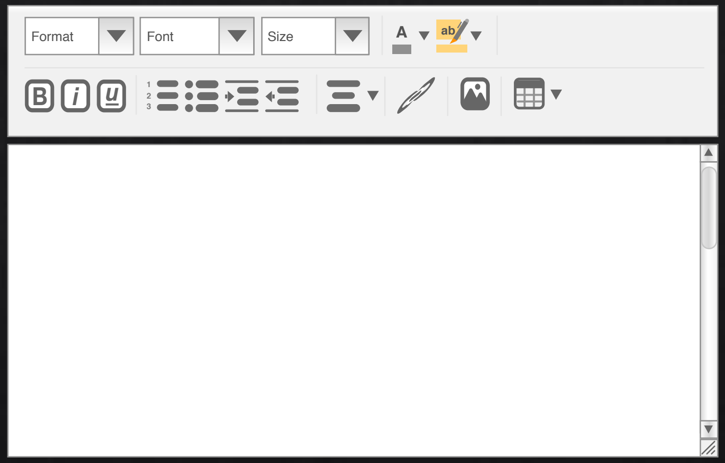
constraint

a relationship that is declared once and automatically maintained

constraint

a relationship that is declared once and automatically maintained

"the toolbar is displayed above the workspace"



constraint

a relationship that is declared once and automatically maintained

“the toolbar is displayed above the workspace”

```
toolbar.x <= workspace.x
toolbar.y <= workspace.y - toolbar.height
```

constraint

a relationship that is declared once and automatically maintained

can produce clearer code by reducing the burden of maintaining relationships

...but not expressive enough for many nuanced behaviors

constraint

a relationship that is declared once and automatically maintained

libraries for imperative ("traditional") languages

e.g. FlapJax, Angular

used for data bindings and specifying layout

[Meyerovich, 2009, angularjs.org]

- “when the toolbar is docked, it is displayed above the workspace”

- “when the toolbar is being dragged, it follows the mouse”

- “when the toolbar is docked, it is displayed above the workspace”
- “when the toolbar is being dragged, it follows the mouse”

state

the status of an interface at a given moment, often controls appearance & behavior

[Letondal 2010, Samek 2003]

state machines

track an interface’s state & when it transitions between states

state machines

track an interface’s state & when it transitions between states

state machines

track an interface's state & when it transitions between states

not_underlined

(clicking adds underline)

state machines

track an interface's state & when it transitions between states

not_underlined

(clicking adds underline)

state machines

track an interface's state & when it transitions between states

not_underlined

(clicking adds underline)

underlined

(remove underline when clicked)

state machines

track an interface’s state & when it transitions between states

not_underlined

(clicking adds underline)

underlined

(remove underline when clicked)

state machines

track an interface's state & when it transitions between states

disabled

(do nothing when clicked)

not_underlined

(clicking adds underline)

underlined

(remove underline when clicked)

state machines

track an interface's state & when it transitions between states

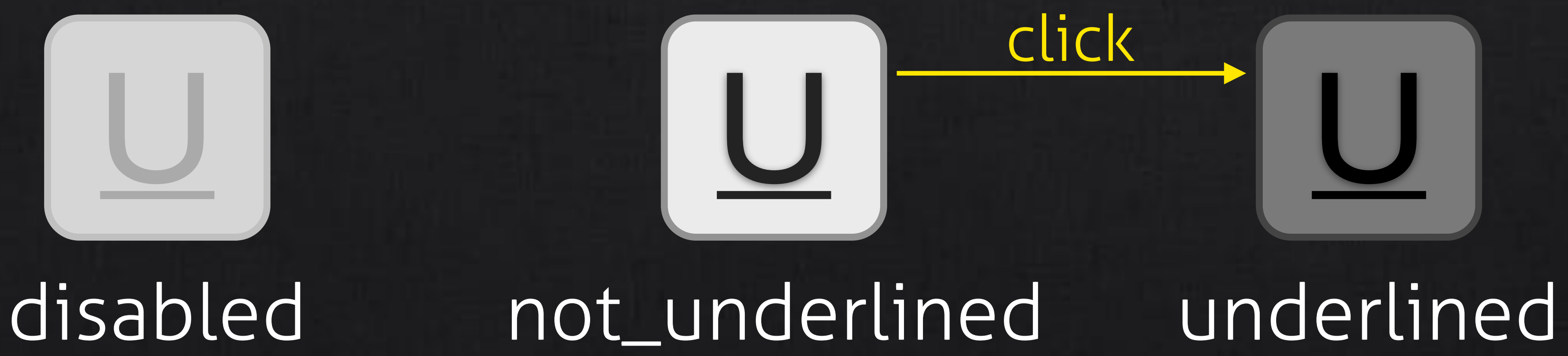
disabled

not_underlined

underlined

state machines

track an interface's state & when it transitions between states



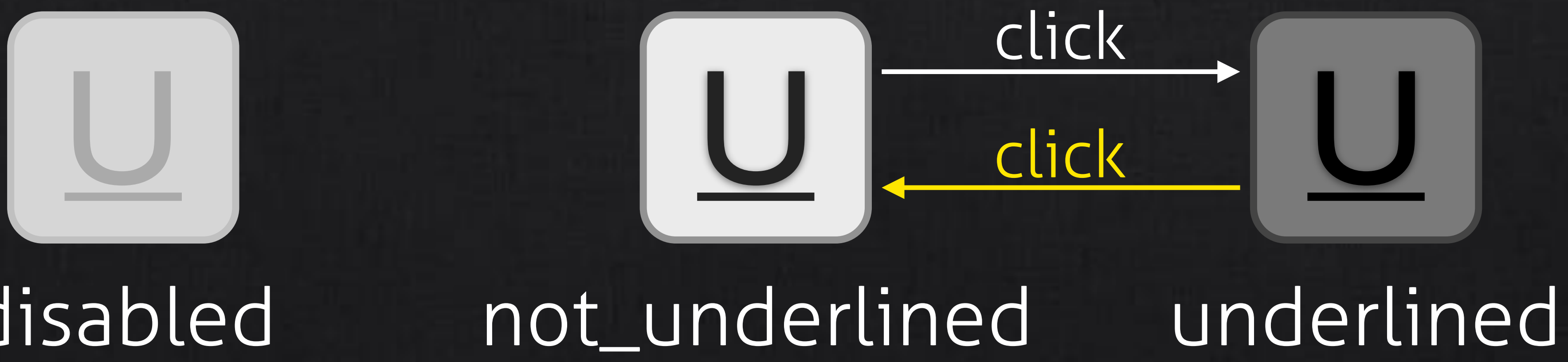
disabled

not_underlined

underlined

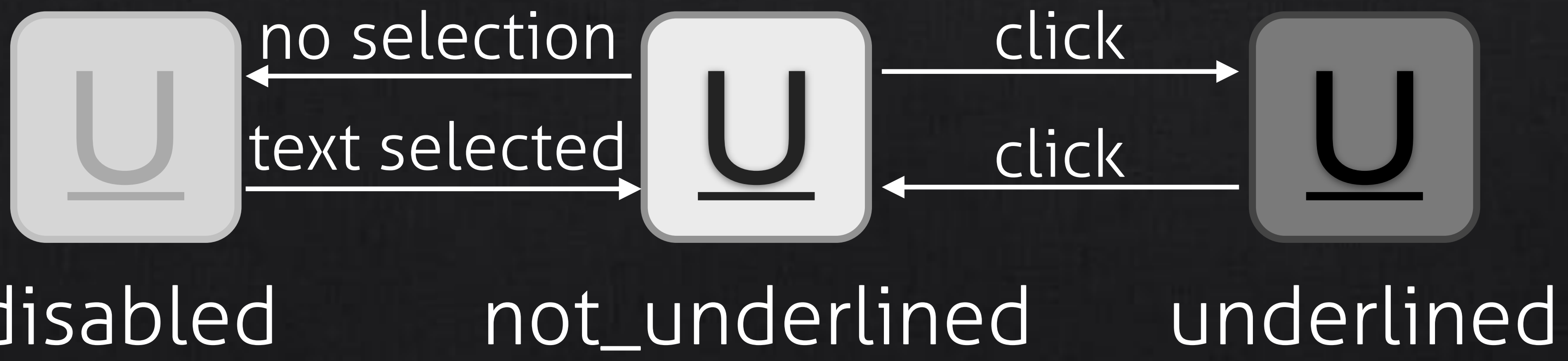
state machines

track an interface's state & when it transitions between states



state machines

track an interface's state & when it transitions between states



state

- GUIs are state-oriented
  - appearance & behavior
- libraries for imperative languages (C, Java, JavaScript, etc.)
  - e.g. SwingStates, SproutCore

[Letondal 2010, Samek 2003, Apperat 2008, sproutcore.com]

- “when the toolbar is docked, it is displayed above the workspace”

- “when the toolbar is being dragged, it follows the mouse”

🟦 state
🟨 constraint

- “when the toolbar is docked, it is displayed above the workspace”
- “when the toolbar is being dragged, it follows the mouse”

🟦 state
🟨 constraint

---

when (state)

then (constraints)

thesis statement

development tools combining constraints with state can express many nuanced interactive behaviors more clearly than event-callback code

outline

- ConstraintJS (JavaScript library)
- InterState (interactive editor)
    - user evaluation
    - multi-touch primitives
- conclusion

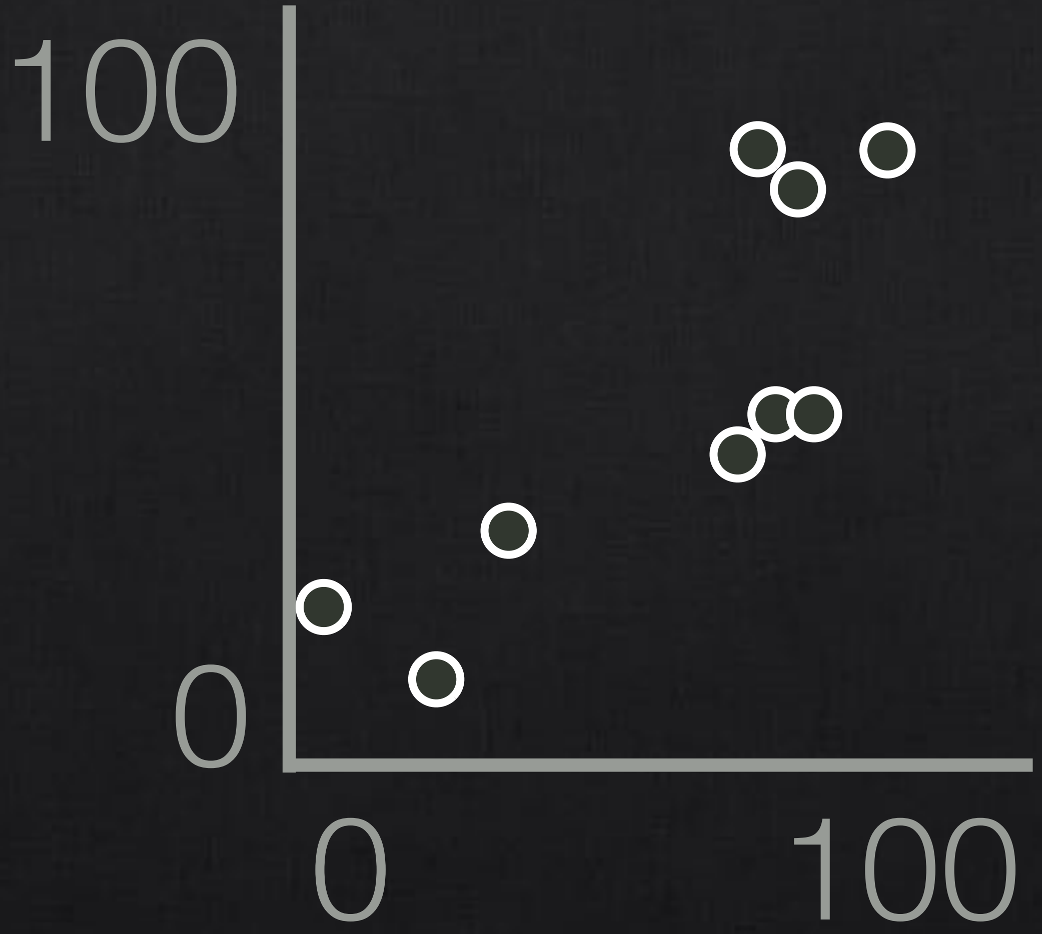
[Oney UIST 2012, Oney UIST 2014]

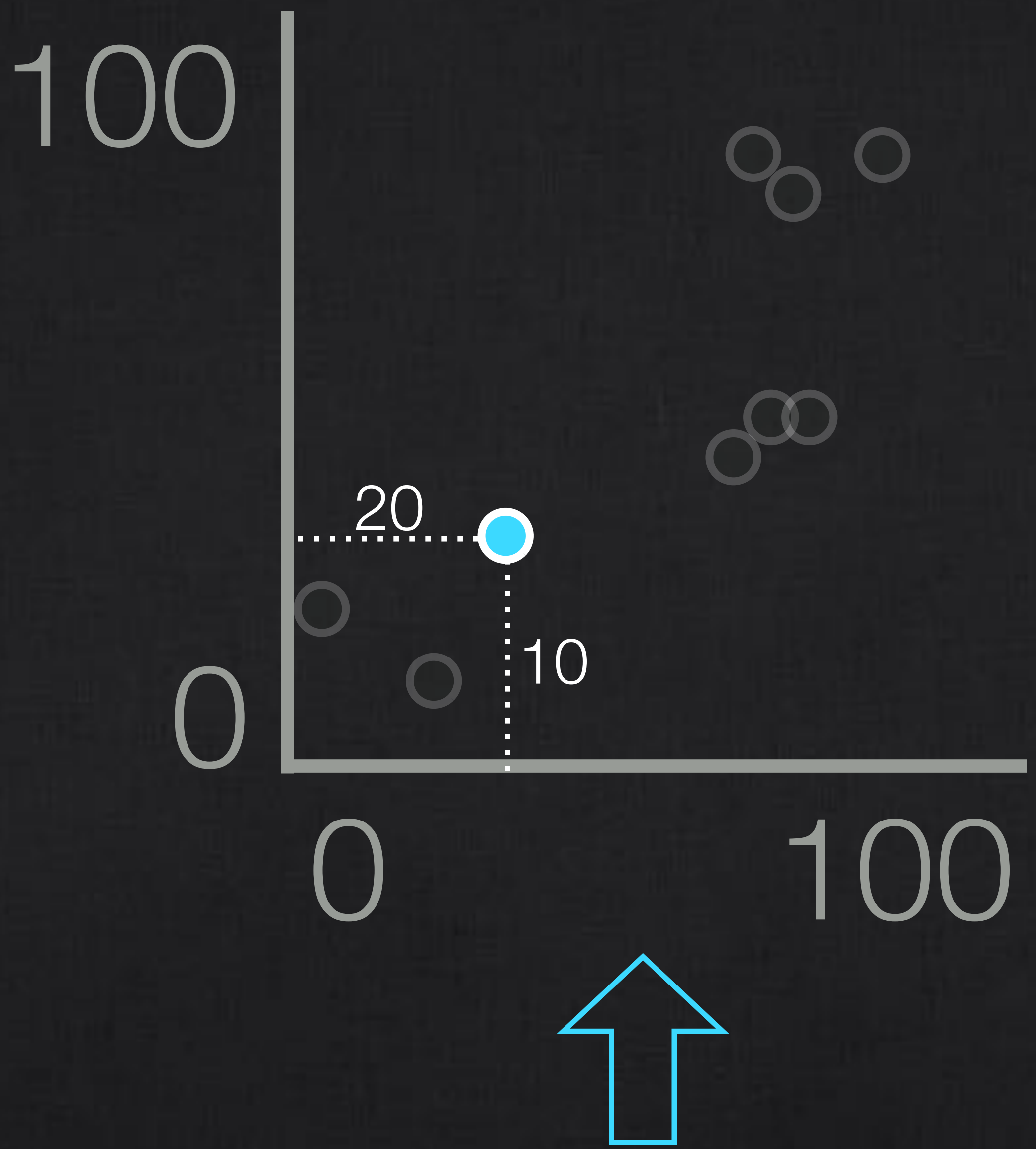
ConstraintJS Contributions

- integrates constraints & states on Web
- integrates with HTML & CSS syntaxes
- efficient implementation

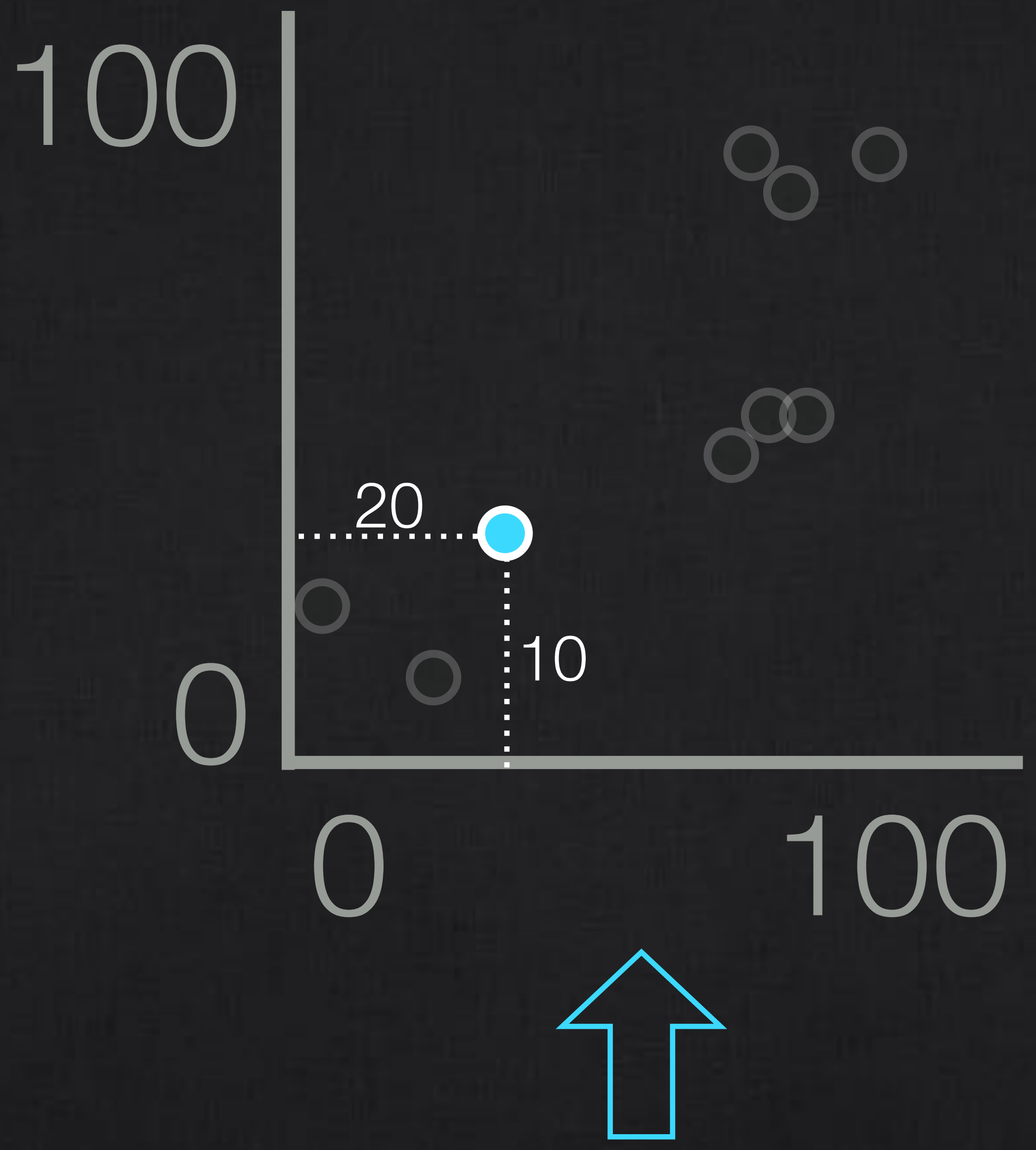
http://cjs.from.so/

[Oney UIST 2012]

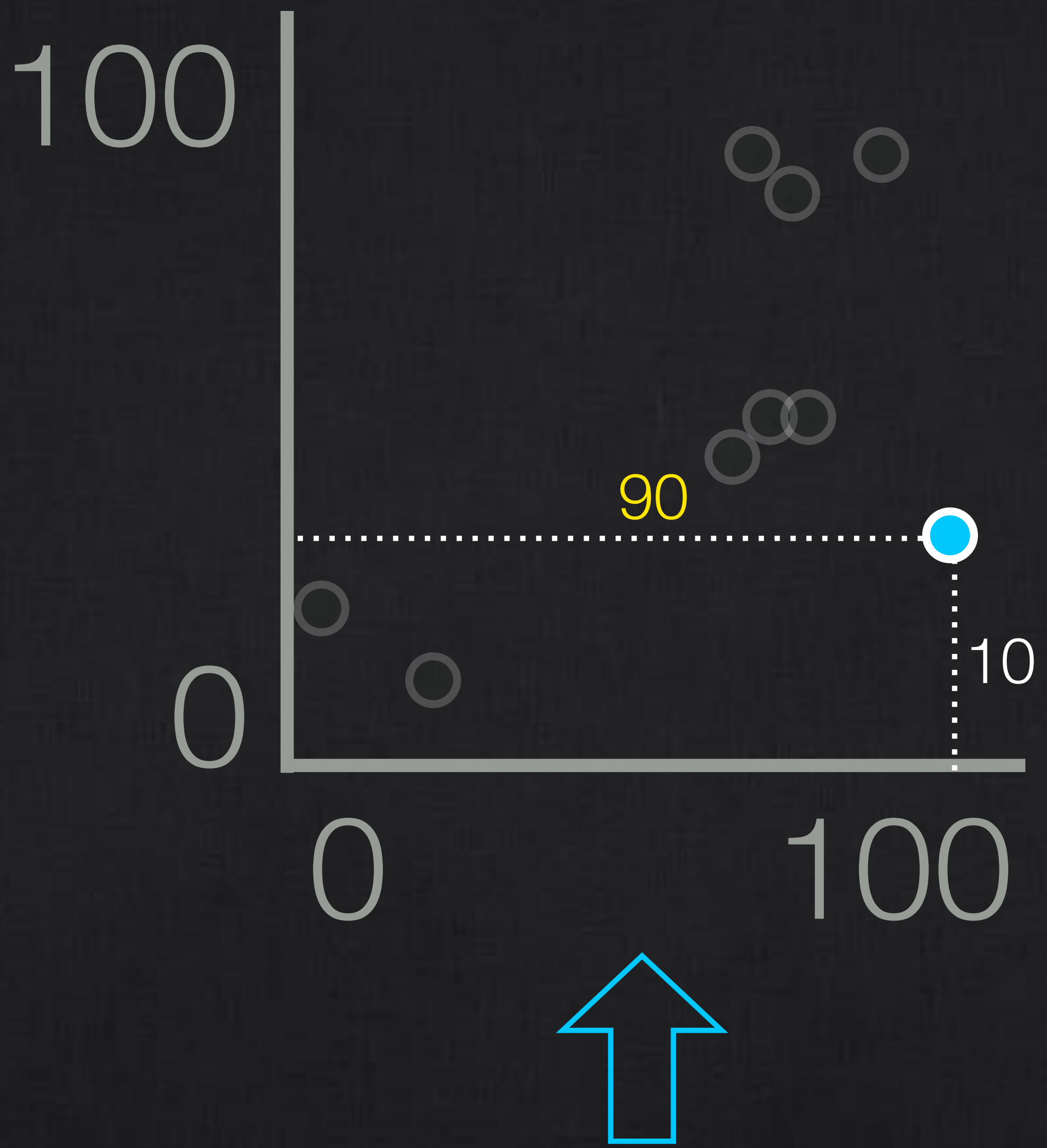

100
0
0
100



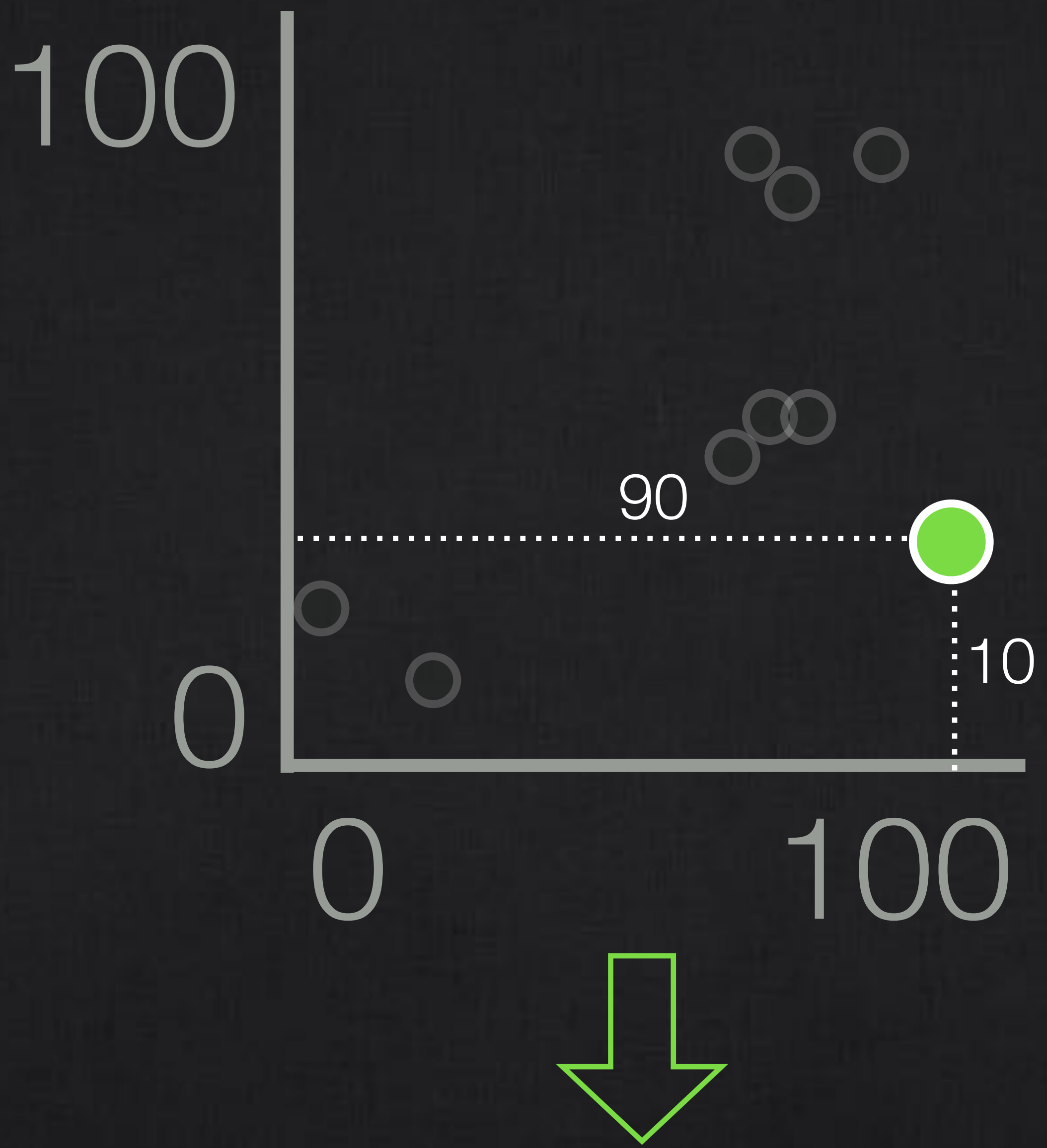
{x:5, y: 20}, {x: 20, y: 10}, {x:30, y:30},
{x:60, y:40}, {x: 65, y: 45}, {x: 70, y: 45},
{x: 63, y: 80}, {x: 68, y: 75}, {x: 80, y: 80}



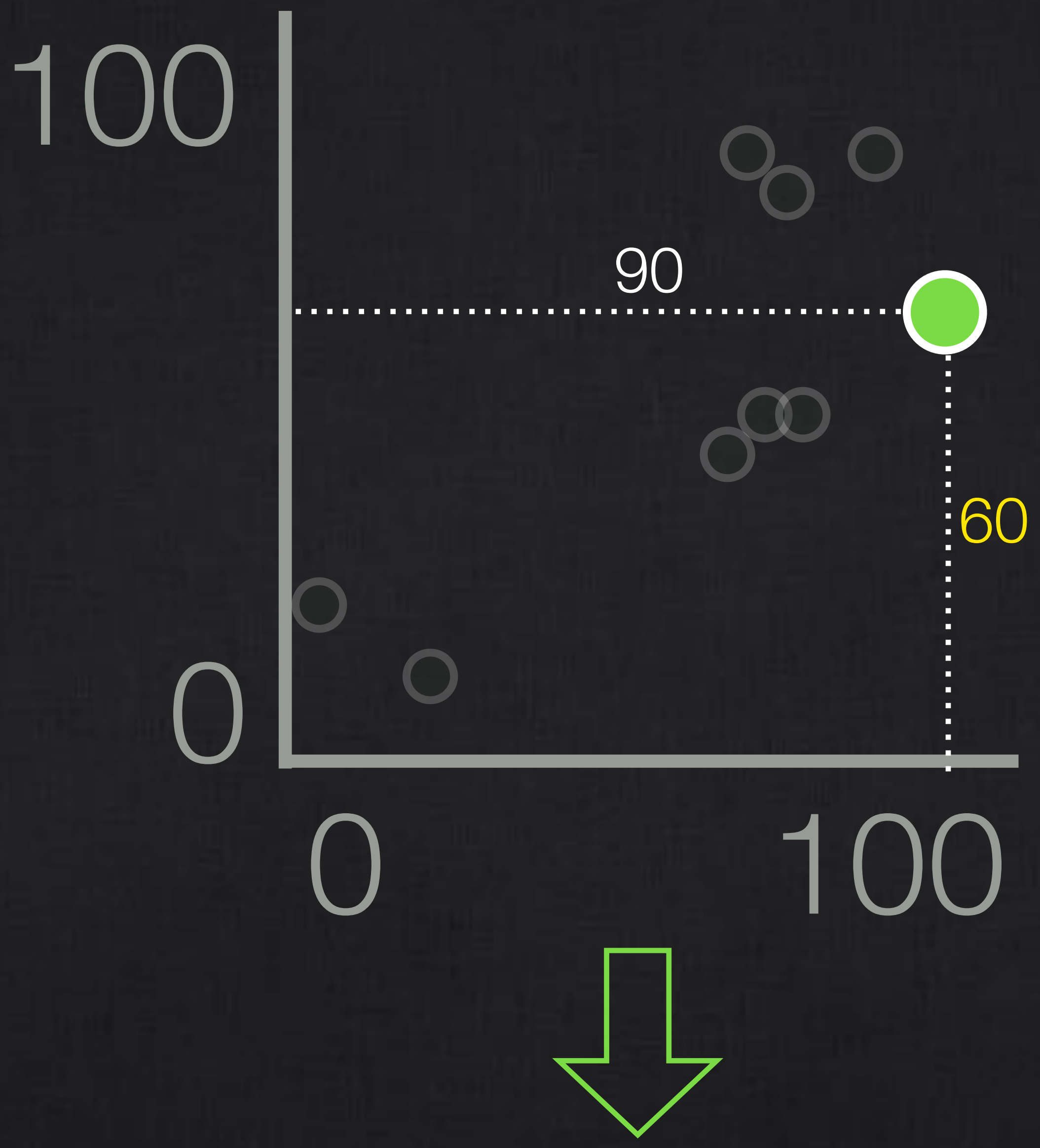
{x:5, y: 20}, {x: 90, y: 10}, {x:30, y:30},
{x:60, y:40}, {x: 65, y: 45}, {x: 70, y: 45},
{x: 63, y: 80}, {x: 68, y: 75}, {x: 80, y: 80}



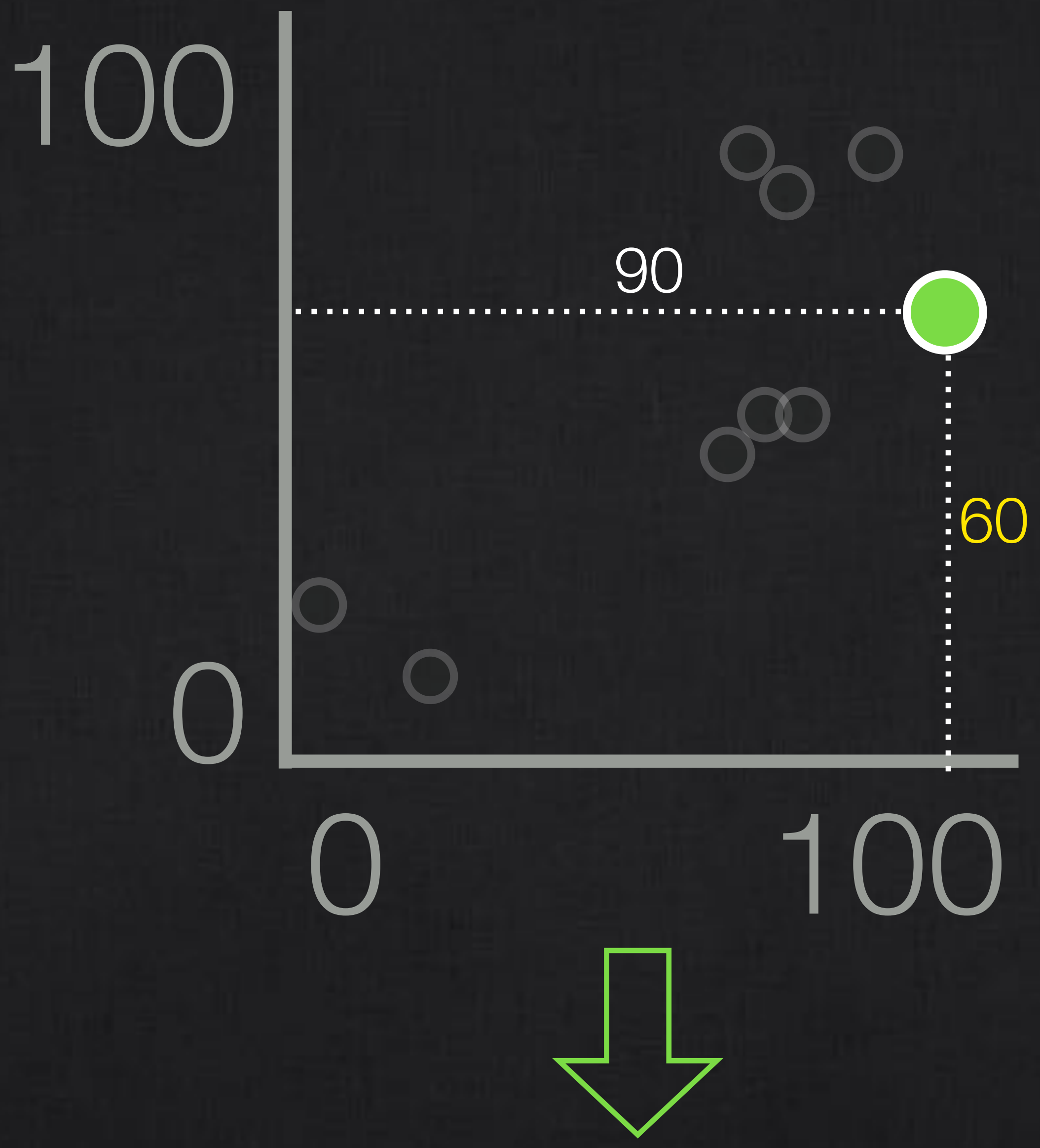
{x:5, y: 20}, {x: 90, y: 10}, {x:30, y:30},
{x:60, y:40}, {x: 65, y: 45}, {x: 70, y: 45},
{x: 63, y: 80}, {x: 68, y: 75}, {x: 80, y: 80}



{x:5, y: 20}, {x: 90, y: 10}, {x:30, y:30},
{x:60, y:40}, {x: 65, y: 45}, {x: 70, y: 45},
{x: 63, y: 80}, {x: 68, y: 75}, {x: 80, y: 80}



{x:5, y: 20}, {x: 90, y: 10}, {x:30, y:30},
{x:60, y:40}, {x: 65, y: 45}, {x: 70, y: 45},
{x: 63, y: 80}, {x: 68, y: 75}, {x: 80, y: 80}



{x:5, y: 20}, {x: 90, y: 60}, {x:30, y:30},
{x:60, y:40}, {x: 65, y: 45}, {x: 70, y: 45},
{x: 63, y: 80}, {x: 68, y: 75}, {x: 80, y: 80}

constraint types

one-way: a<=b+1

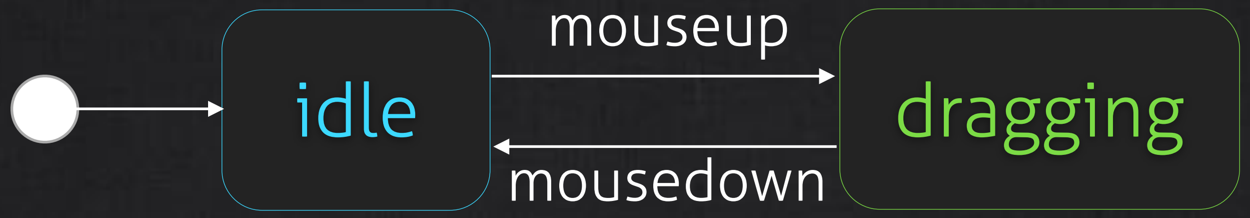
solves for a

multi-way: a<=>b+1

solves for a or b
uses hierarchy (difficult to control)

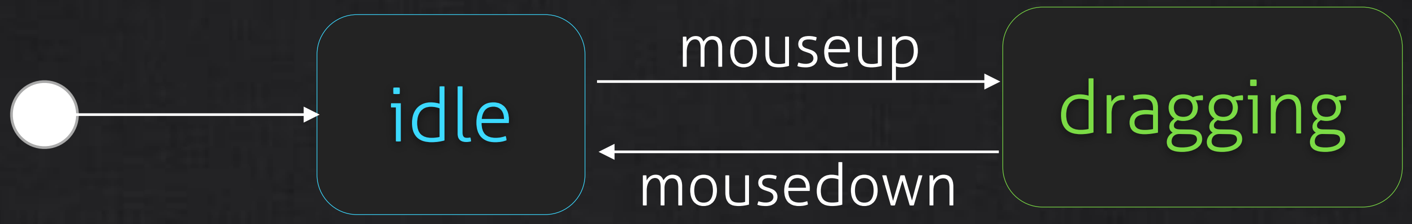
[Vander Zanden 1994]

`fsm =`



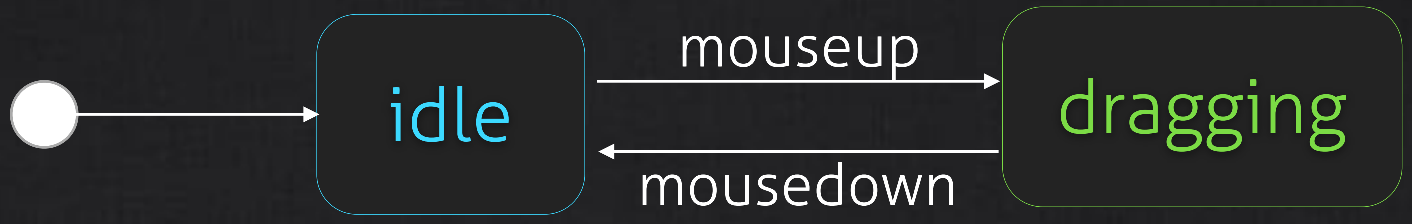
state machine (for every point)

```
fsm =
```



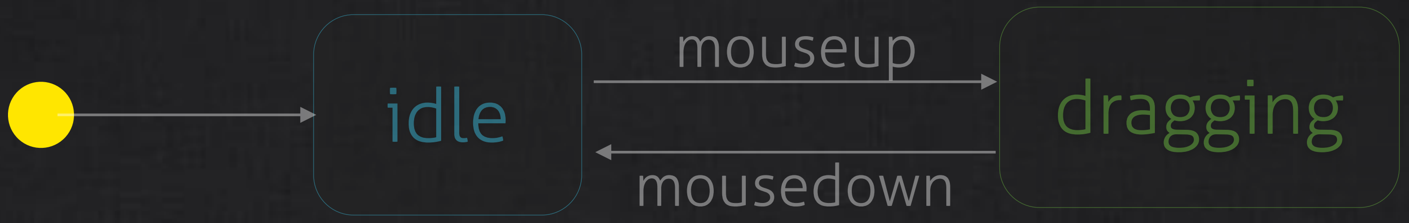
```
 view_x = cjs(fsm, {
                         idle:
                         model_x,      dragging:
                                       cjs.mouse.x
});

model_x = cjs(fsm, {
                 init:
                 datum.x,              dragging:
                                       view_x
});
```

fsm =



```
 view_x = cjs(fsm, {
                          idle:
                          model_x,       dragging:
                                         cjs.mouse.x
});

model_x = cjs(fsm, {
                  init:
                  datum.x,               dragging:
                                         view_x
});
```

```
fsm =
```



idle

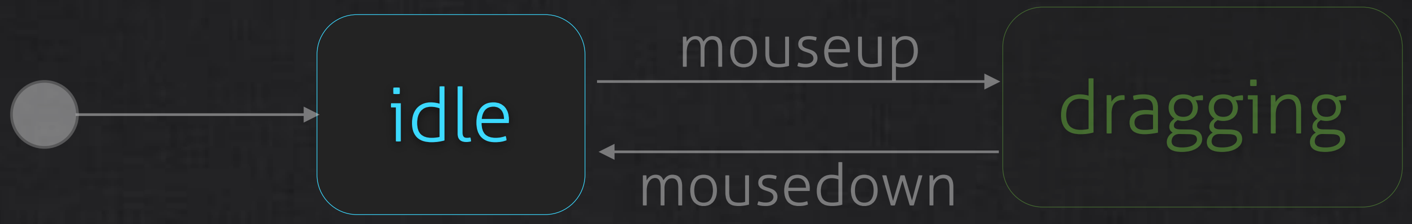
mouseup

mousedown

dragging

```
view_x = cjs(fsm, {
                          idle:
                          model_x,         dragging:
                                           cjs.mouse.x
});

model_x = cjs(fsm, {
                  init:
                  datum.x,                 dragging:
                                           view_x
});
```

```
fsm =
```


idle

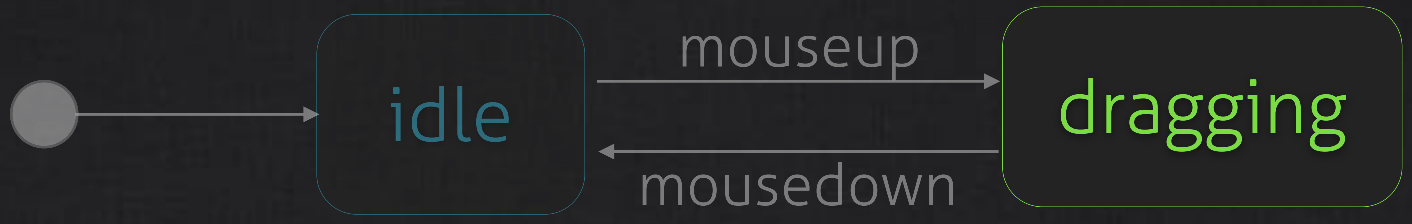
mouseup

mousedown

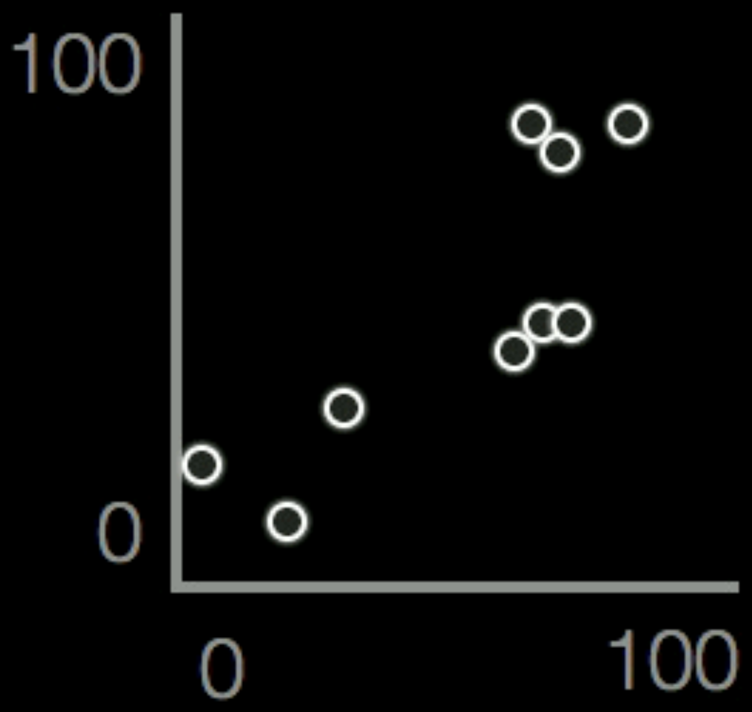
dragging

```
view_x = cjs(fsm, {
                     idle:               dragging:
                     model_x,            cjs.mouse.x
});

model_x = cjs(fsm, {
               init:                     dragging:
               datum.x,                  view_x
});
```

fsm =



```
view_x = cjs(fsm, {
                    idle:
                    model_x,        dragging:
                                    cjs.mouse.x
});

model_x = cjs(fsm, {
                 init:
                 datum.x,           dragging:
                                    view_x
});
```

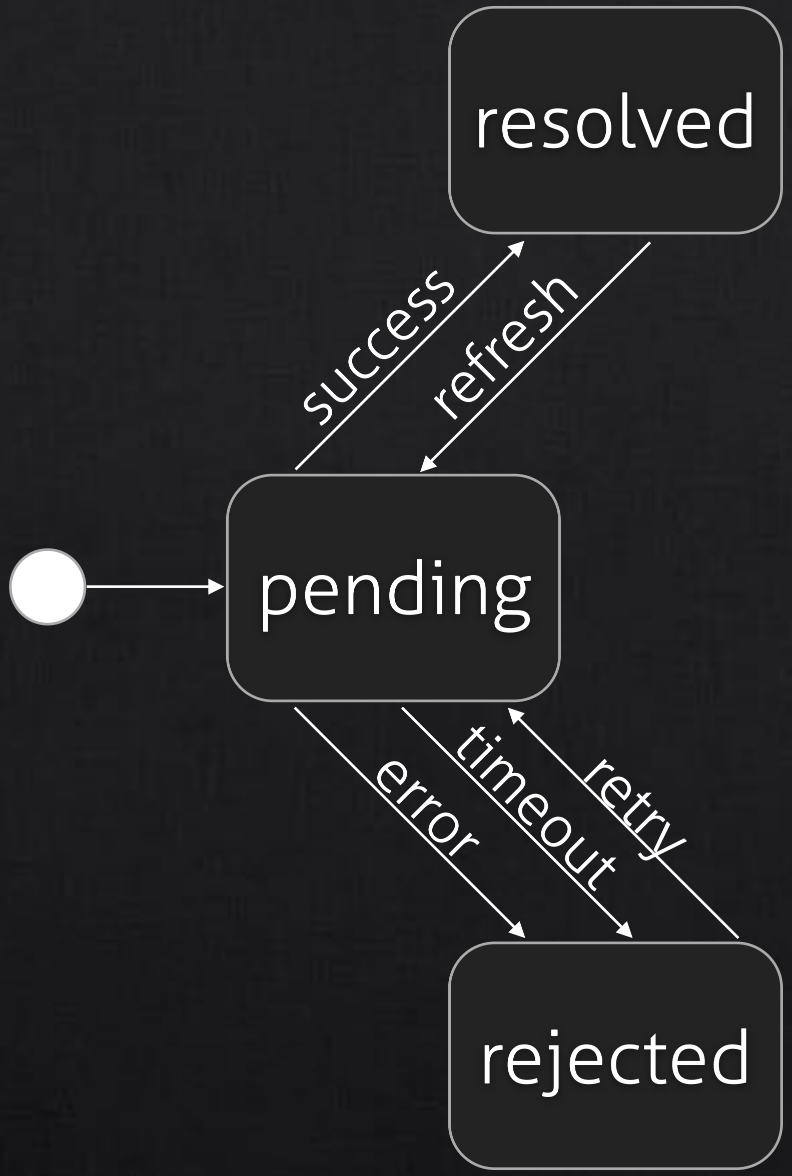

100
0
0
100

asynchronous values

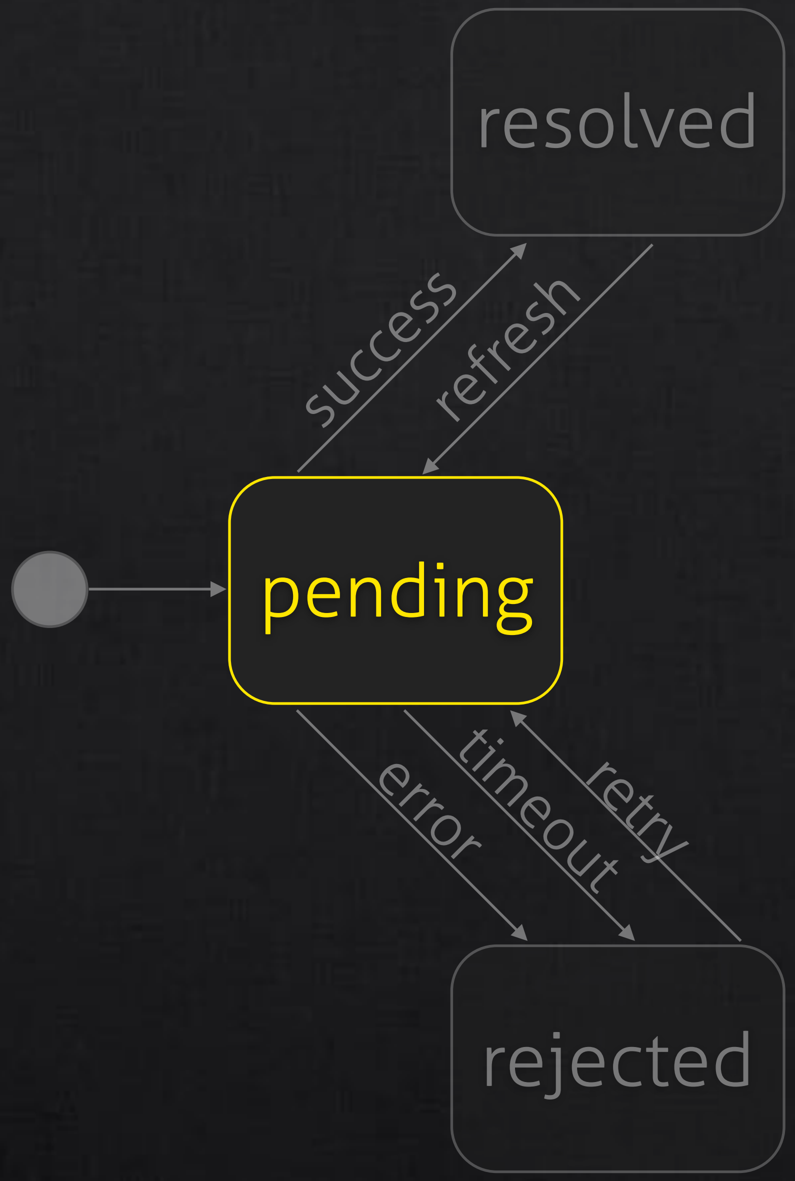
- indeterminate wait before return
- can greatly increase complexity of imperative code
    - control timing
    - propagation of values

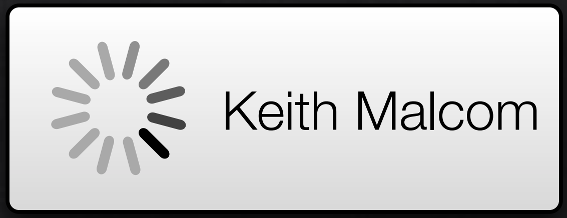
Loading friends...

asynchronous values



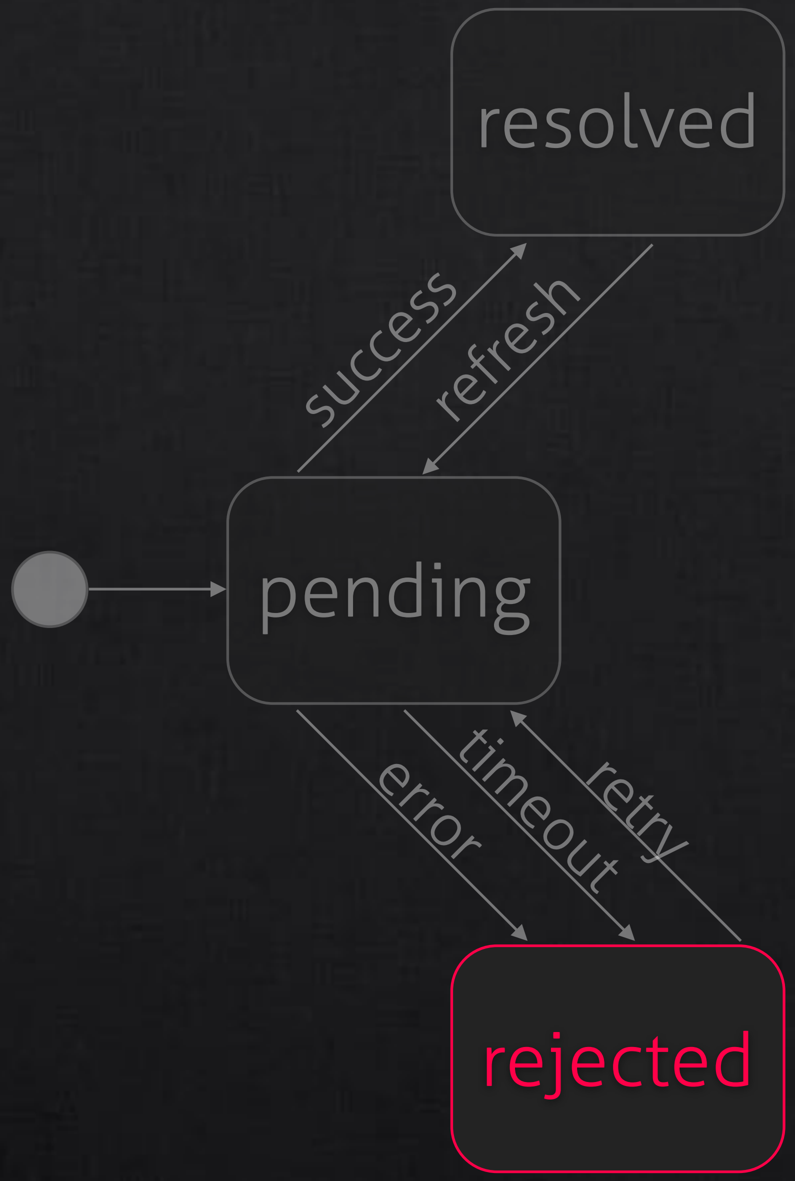
asynchronous values

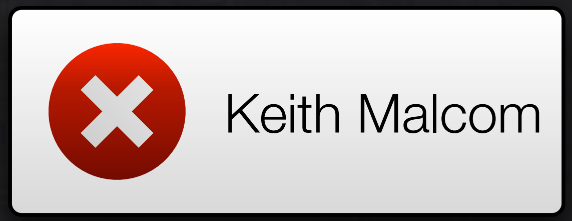




```
pending:   "loading.gif"
```

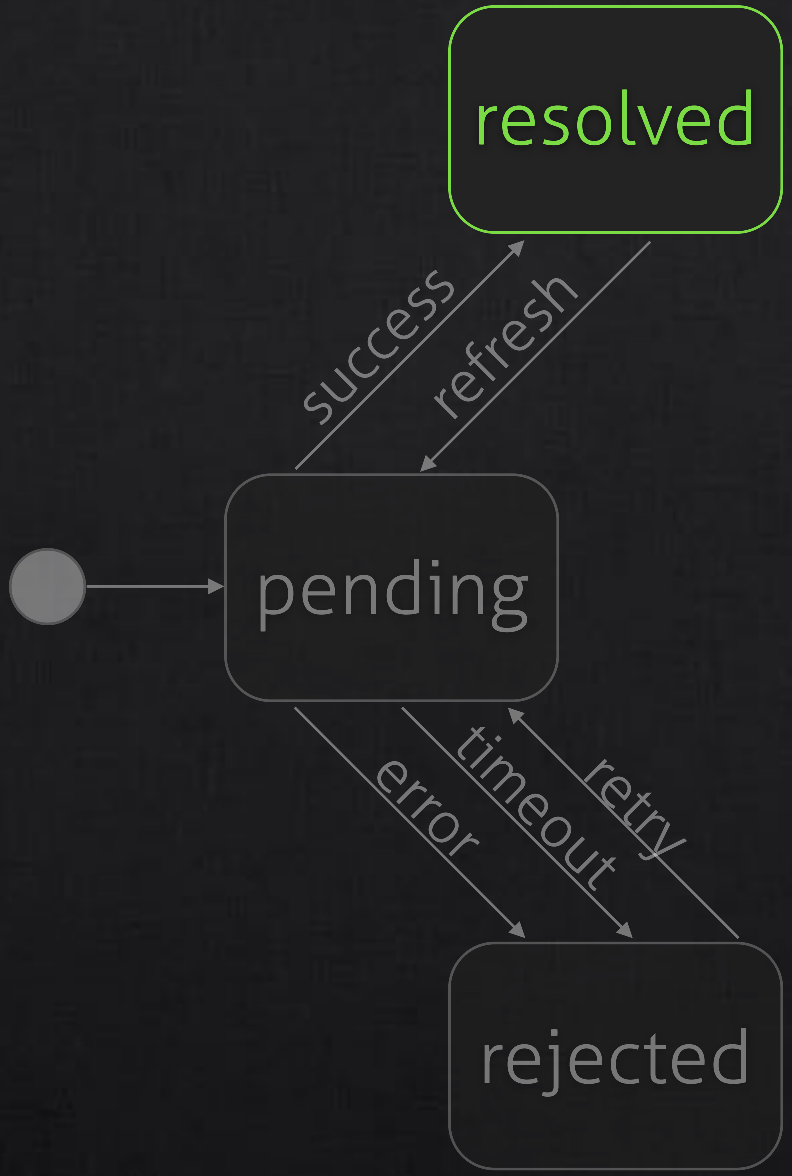
asynchronous values





```
pending:  "loading.gif",
rejected: "error.gif"
```

asynchronous values



```
pending:  "loading.gif",
rejected: "error.gif"   ,
resolved: {{ picture }}
```

```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                               + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                               + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

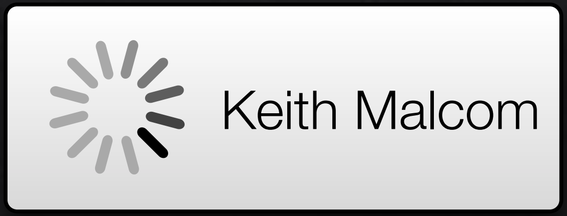
```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                             + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                               + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

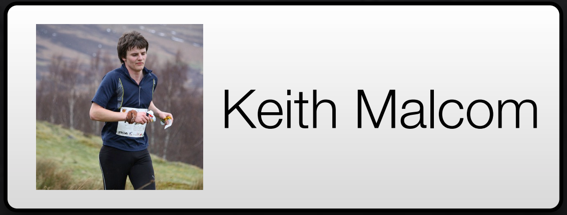
```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                               + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                               + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

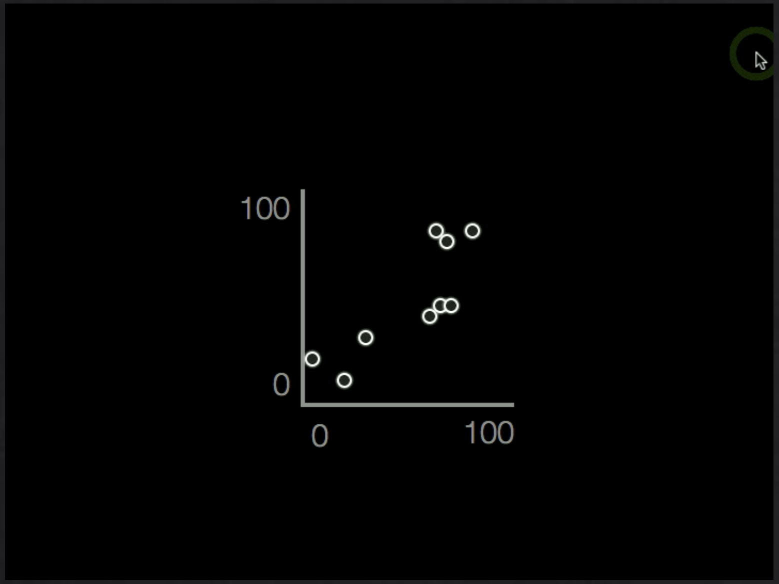
```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                             + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```



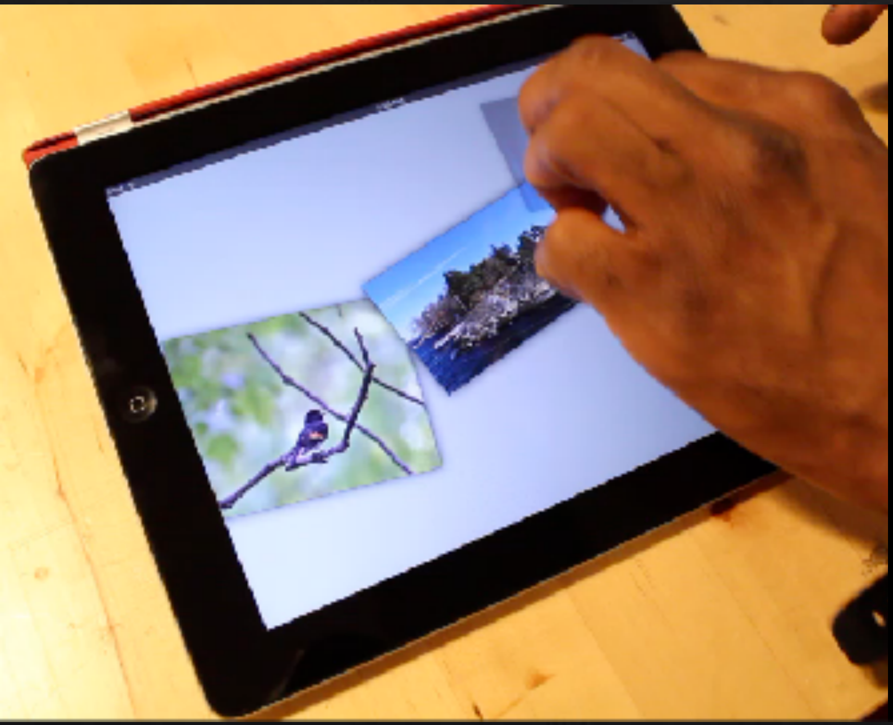
```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                               + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```

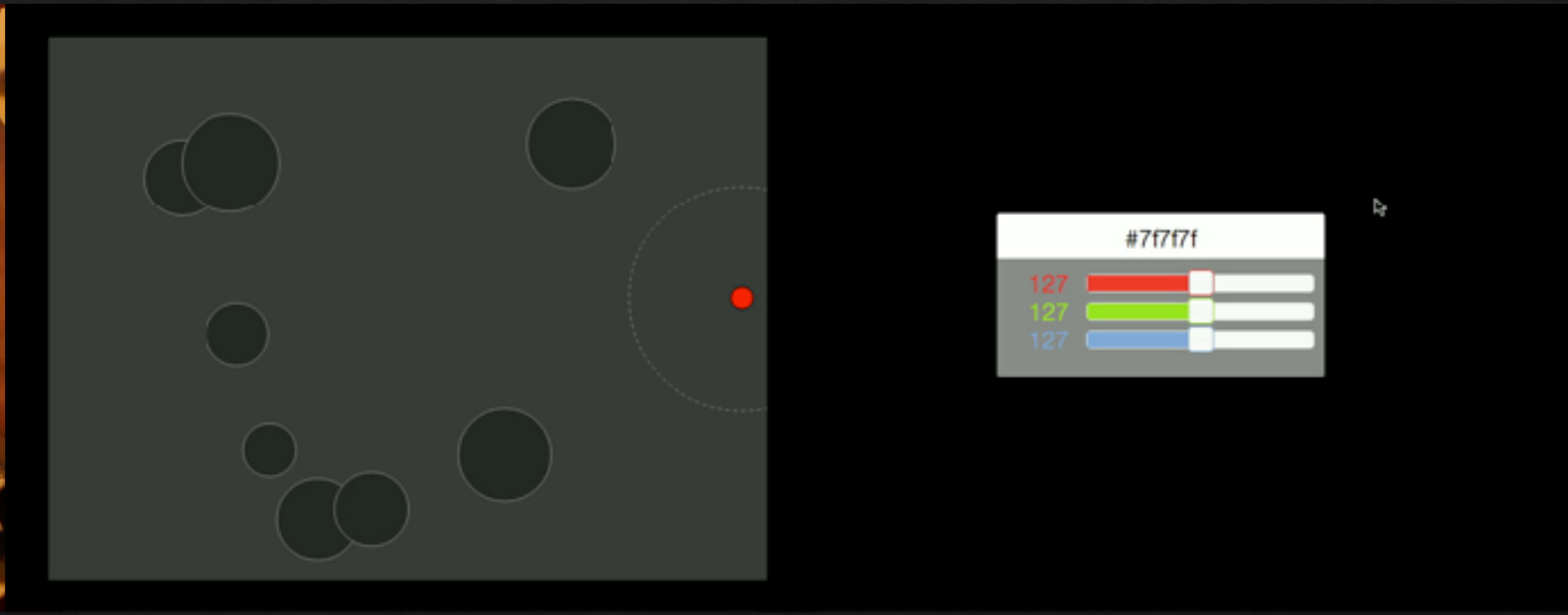


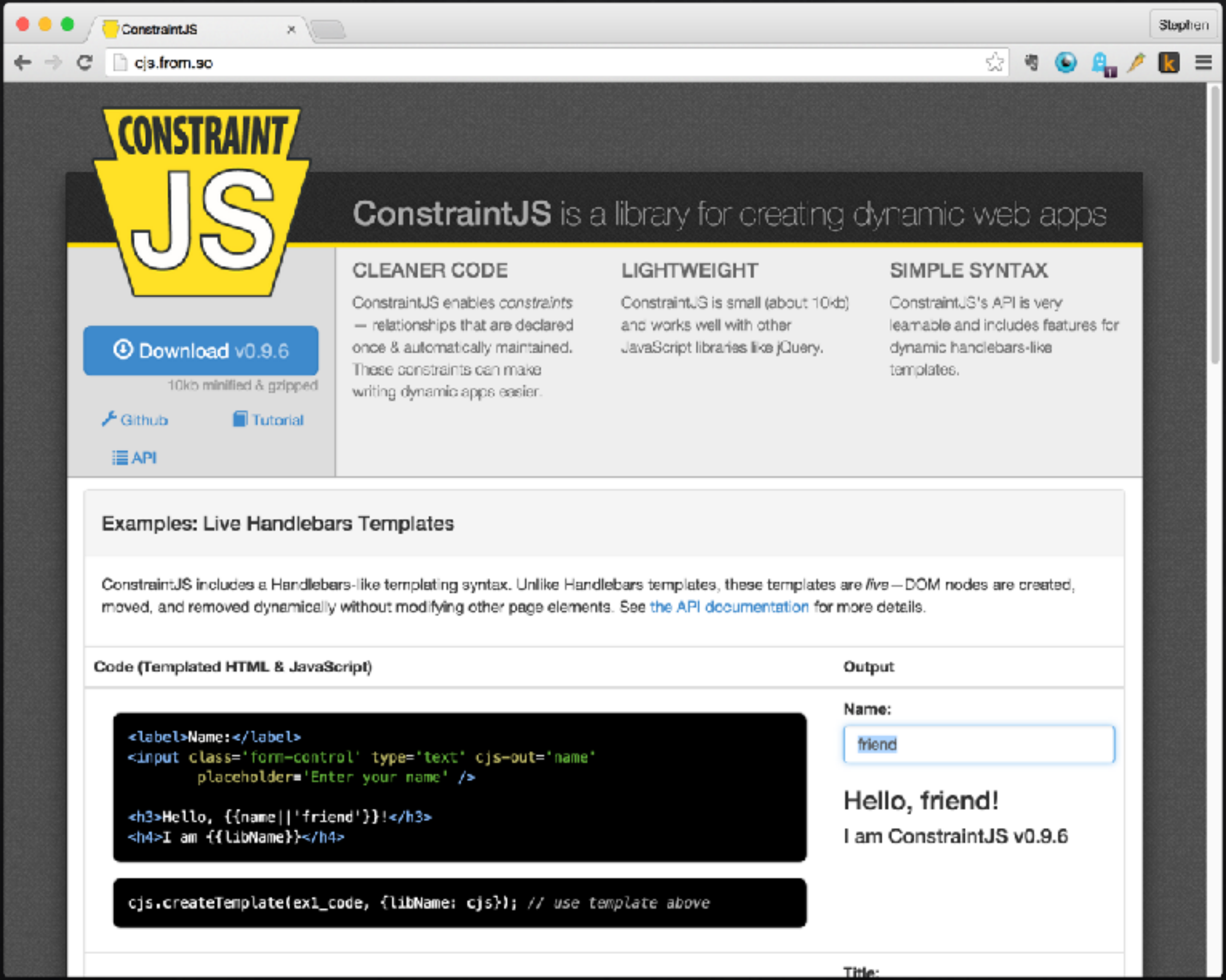
```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request("/" + friend.id
                                              + “/picture"));
          });

// Reviewer’s code below:
load_friends = (selector) ->
     people = $(selector).text("Loading friends...")
     FB.api "/me/friends", (answer) ->
        if friends=answer.data
           for friend in friends
              people.append $("<div>#{friend.name}</div>")
                    .prepend(img=$('<img/>'))
              img.attr "src", "loading.gif"
              FB.api "/#{friend.id}/picture", (answer) ->
                 if picture=answer.data
                    img.attr "src", picture
                 else
                    img.attr "src", "error.gif"
        else
           friends.text("Error")
```


100
0
0
100

Aruna Balakrishnan
Stephanie Gordon
Robert Young
Logan Oney
Alexi Soltt
Abe Othman
Joy Nixon
Chevy Dixon
Byron McLain
Kenyatta Craten
Derrick Barker
Christie Auw
Michael Bernstein
Field Cady
Nicole Fernandez
Tony Galhano
Brian Lim
Brandon Caestin
Natalie Wilson
Unwana Otu
Jesus Alvarez




#7f7f7f
127
127
127



ConstraintJS is a library for creating dynamic web apps

CLEANER CODE

ConstraintJS enables *constraints* — relationships that are declared once & automatically maintained. These constraints can make writing dynamic apps easier.

LIGHTWEIGHT

ConstraintJS is small (about 10kb) and works well with other JavaScript libraries like jQuery.

SIMPLE SYNTAX

ConstraintJS's API is very learnable and includes features for dynamic handlebars-like templates.

Examples: Live Handlebars Templates

ConstraintJS includes a Handlebars-like templating syntax. Unlike Handlebars templates, these templates are *live* — DOM nodes are created, moved, and removed dynamically without modifying other page elements. See the API documentation for more details.

Code (Templated HTML & JavaScript)

```
<label>Name:</label>
<input class='form-control' type='text' cjs-out='name'
       placeholder='Enter your name' />

<h3>Hello, {{name||'friend'}}!</h3>
<h4>I am {{libName}}</h4>
```

```
cjs.createTemplate(ex1_code, {libName: cjs}); // use template above
```

Output

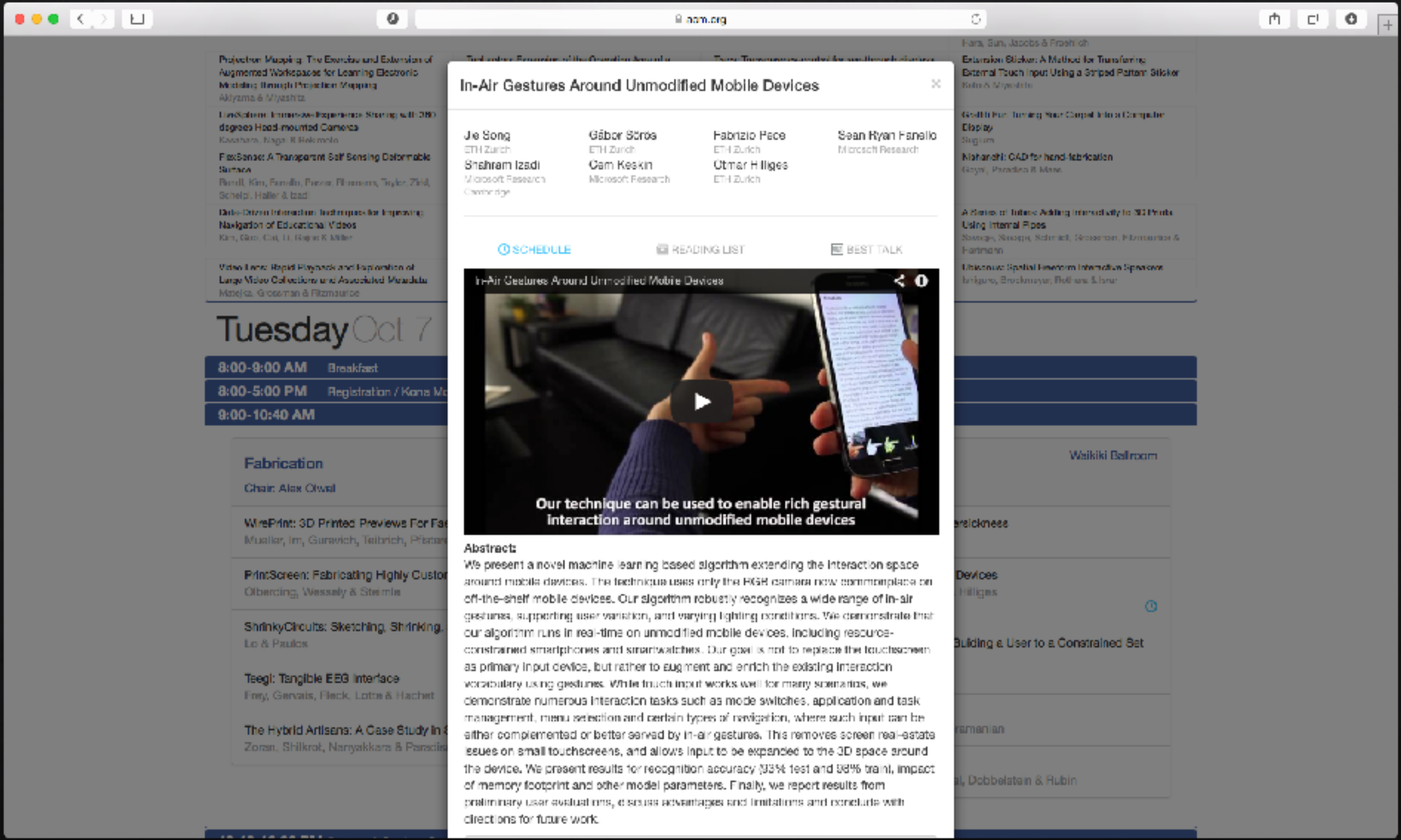
Name: friend

Hello, friend!

I am ConstraintJS v0.9.6

Title:

http://cjs.from.so/



acm.org

Projection Mapping: The Exercise and Extension of Augmented Workspaces for Learning Electronic Modeling through Projection Mapping
Akiyama & Miyashita

LiveSphere: Immersive Experience Sharing with 360 degrees Head-mounted Cameras
Kasahara, Nagai & Rekimoto

FlexSense: A Transparent Self-Sensing Deformable Surface
Rendl, Kim, Fanello, Parzer, Rhemann, Taylor, Zirkl, Scheipl, Haller & Izadi

Data-Driven Interaction Techniques for Improving Navigation of Educational Videos
Kim, Guo, Cai, Li, Gajos & Miller

Video Lens: Rapid Playback and Exploration of Large Video Collections and Associated Metadata
Matejka, Grossman & Fitzmaurice

Fara, Sun, Jacobs & Froehlich

Extension Sticker: A Method for Transferring External Touch Input Using a Striped Pattern Sticker
Kato & Miyashita

Graffiti Fur: Turning Your Carpet into a Computer Display
Sugiura

Nishanchi: CAD for hand-fabrication
Goyal, Paradiso & Maes

A Series of Tubes: Adding Interactivity to 3D Prints Using Internal Pipes
Savage, Schmidt, Grossman, Fitzmaurice & Hartmann

[illegible] Spatial Freeform Interaction Speakers
Ishiguro, Brockmeyer, [illegible] & Israr

Tuesday Oct 7

8:00-9:00 AM Breakfast

8:00-5:00 PM Registration / Kona M

9:00-10:40 AM

Fabrication
Chair: Alex Olwal
Waikiki Ballroom

WirePrint: 3D Printed Previews For Fa
Mueller, Im, Gurevich, Teibrich, Pfisterer

PrintScreen: Fabricating Highly Custo
Olberding, Wessely & Steimle

ShrinkyCircuits: Sketching, Shrinking,
Lo & Paulos

Teegi: Tangible EEG Interface
Frey, Gervais, Fleck, Lotte & Hachet

The Hybrid Artisans: A Case Study in
Zoran, Shilkrot, Nanyakkara & Paradis

erickness

Devices
Hilliges

Building a User to a Constrained Set

ramanian

al, Dobbelstein & Rubin

In-Air Gestures Around Unmodified Mobile Devices

Jie Song
ETH Zurich

Gábor Sörös
ETH Zurich

Fabrizio Pece
ETH Zurich

Sean Ryan Fanello
Microsoft Research

Shahram Izadi
Microsoft Research Cambridge

Cem Keskin
Microsoft Research

Otmar Hilliges
ETH Zurich

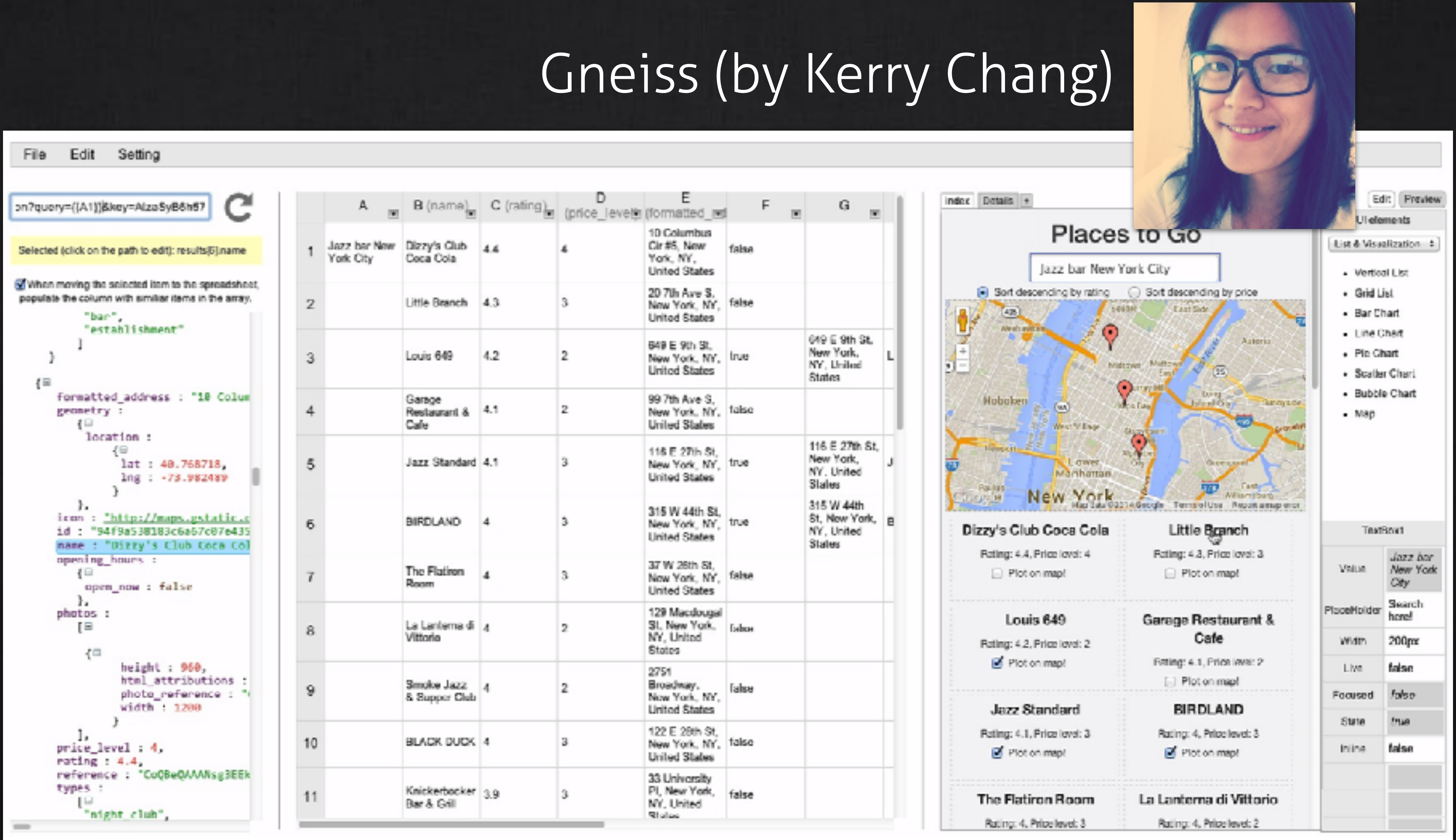
SCHEDULE
READING LIST
BEST TALK
In-Air Gestures Around Unmodified Mobile Devices
Our technique can be used to enable rich gestural interaction around unmodified mobile devices

Abstract:

We present a novel machine learning based algorithm extending the interaction space around mobile devices. The technique uses only the RGB camera now commonplace on off-the-shelf mobile devices. Our algorithm robustly recognizes a wide range of in-air gestures, supporting user variation, and varying lighting conditions. We demonstrate that our algorithm runs in real-time on unmodified mobile devices, including resource-constrained smartphones and smartwatches. Our goal is not to replace the touchscreen as primary input device, but rather to augment and enrich the existing interaction vocabulary using gestures. While touch input works well for many scenarios, we demonstrate numerous interaction tasks such as mode switches, application and task management, menu selection and certain types of navigation, where such input can be either complemented or better served by in-air gestures. This removes screen real-estate issues on small touchscreens, and allows input to be expanded to the 3D space around the device. We present results for recognition accuracy (93% test and 98% train), impact of memory footprint and other model parameters. Finally, we report results from preliminary user evaluations, discuss advantages and limitations and conclude with directions for future work.

CHI & UIST Interactive Web Programs

Gneiss (by Kerry Chang)



on?query=([A1])&key=AIzaSyB6h57

Selected (click on the path to edit): results[6].name

☑ When moving the selected item to the spreadsheet, populate the column with similar items in the array.

```
            "bar",
            "establishment"
         ]
   }

  {⊟
     formatted_address : "10 Colum
     geometry :
        {⊟
         location :
            {⊟
             lat : 40.768718,
             lng : -73.982489
            }
        },
     icon : "http://maps.gstatic.c
     id : "94f9a538183c6a67c07e435
     name : "Dizzy's Club Coca Col
     opening_hours :
        {⊟
         open_now : false
        },
     photos :
        [⊟

         {⊟
             height : 960,
             html_attributions :
             photo_reference : "
             width : 1280
            }
        ],
     price_level : 4,
     rating : 4.4,
     reference : "CoQBeQAAANsg3EEk
     types :
        [⊟
         "night_club",
```

	A	B (name)	C (rating)	D (price_level)	E (formatted_ad	F	G	
1	Jazz bar New York City	Dizzy's Club Coca Cola	4.4	4	10 Columbus Cir #5, New York, NY, United States	false		
2		Little Branch	4.3	3	20 7th Ave S, New York, NY, United States	false		
3		Louis 649	4.2	2	649 E 9th St, New York, NY, United States	true	649 E 9th St, New York, NY, United States	L
4		Garage Restaurant & Cafe	4.1	2	99 7th Ave S, New York, NY, United States	false		
5		Jazz Standard	4.1	3	116 E 27th St, New York, NY, United States	true	116 E 27th St, New York, NY, United States	J
6		BIRDLAND	4	3	315 W 44th St, New York, NY, United States	true	315 W 44th St, New York, NY, United States	B
7		The Flatiron Room	4	3	37 W 26th St, New York, NY, United States	false		
8		La Lanterna di Vittorio	4	2	129 Macdougal St, New York, NY, United States	false		
9		Smoke Jazz & Supper Club	4	2	2751 Broadway, New York, NY, United States	false		
10		BLACK DUCK	4	3	122 E 28th St, New York, NY, United States	false		
11		Knickerbocker Bar & Grill	3.9	3	33 University Pl, New York, NY, United States	false		

Places to Go

Jazz bar New York City

● Sort descending by rating ○ Sort descending by price

Dizzy's Club Coca Cola
Rating: 4.4, Price level: 4
□ Plot on map!

Little Branch
Rating: 4.3, Price level: 3
□ Plot on map!

Louis 649
Rating: 4.2, Price level: 2
☑ Plot on map!

Garage Restaurant & Cafe
Rating: 4.1, Price level: 2
□ Plot on map!

Jazz Standard
Rating: 4.1, Price level: 3
☑ Plot on map!

BIRDLAND
Rating: 4, Price level: 3
☑ Plot on map!

The Flatiron Room
Rating: 4, Price level: 3

La Lanterna di Vittorio
Rating: 4, Price level: 2

- Vertical List
- Grid List
- Bar Chart
- Line Chart
- Pie Chart
- Scatter Chart
- Bubble Chart
- Map

Value	Jazz bar New York City
PlaceHolder	Search here!
Width	200px
Live	false
Focused	false
State	true
Inline	false

[Chang UIST 2014, Chang CHI 2015]

InterState

- visual notation and live editor
- expressive and understandable

[Oney UIST 2014]

InterState Contributions

Computational Model

- clear & concise
- states & constraints

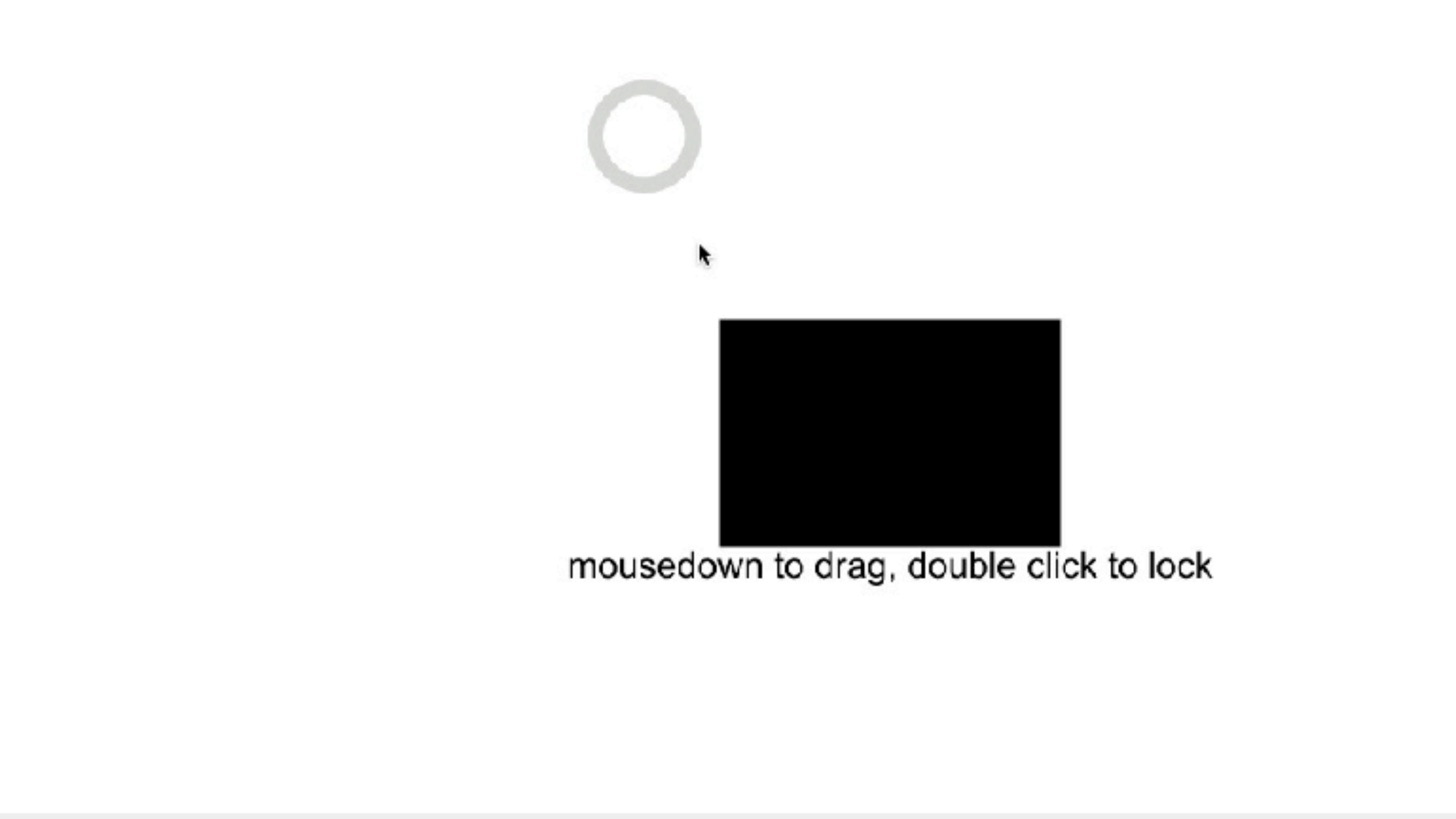
InterState Contributions

Computational Model

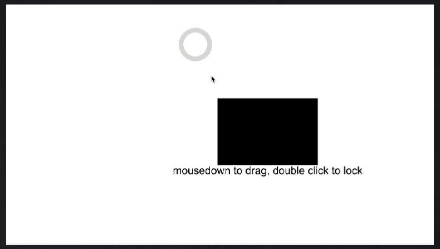
- clear & concise
- states & constraints

Visual Notation

- suits computational model
- states & constraints in grid

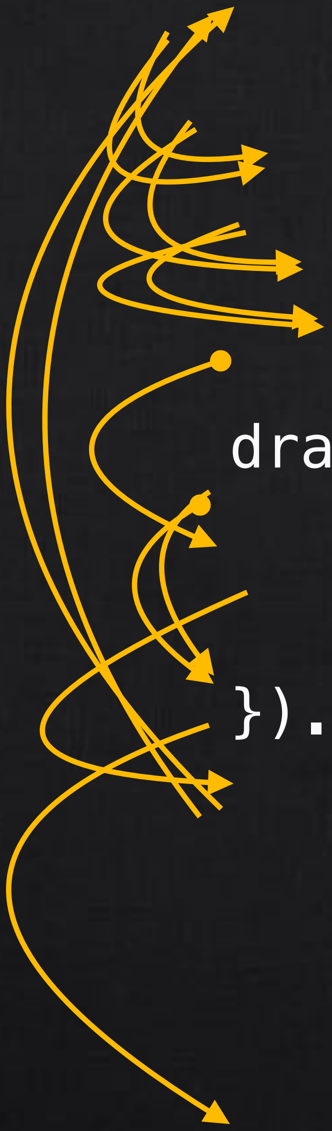


mousedown to drag, double click to lock

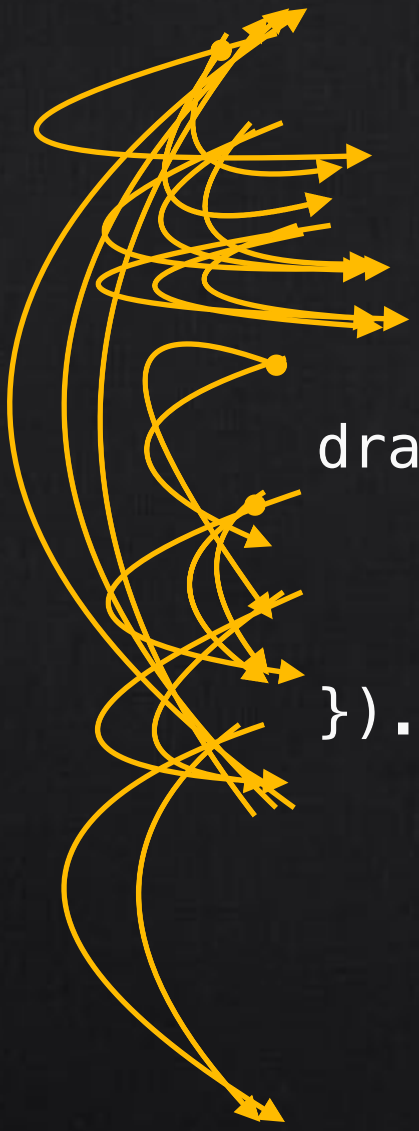


mousedown to drag, double click to lock

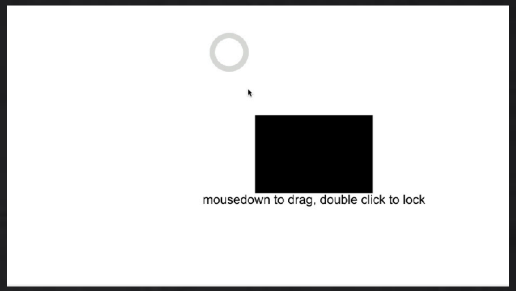
```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
    }
    isDragLocked = !isDragLocked;
});
```

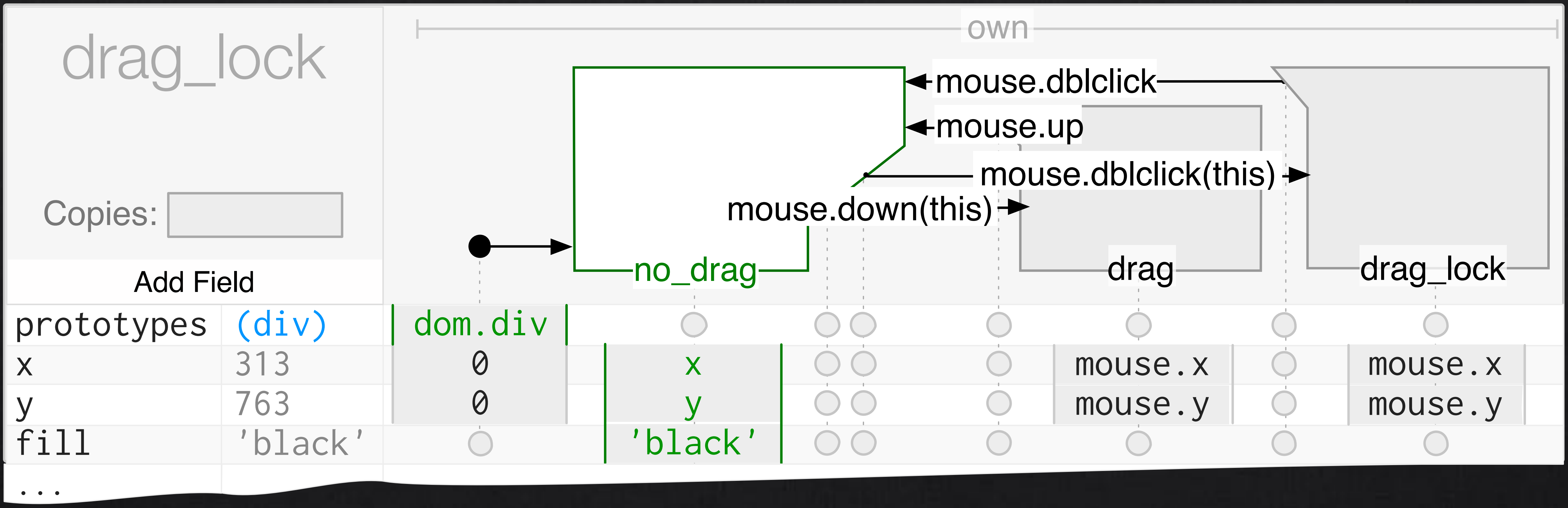


```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
    }
    isDragLocked = !isDragLocked;
});
```



```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
    }
    isDragLocked = !isDragLocked;
});
```



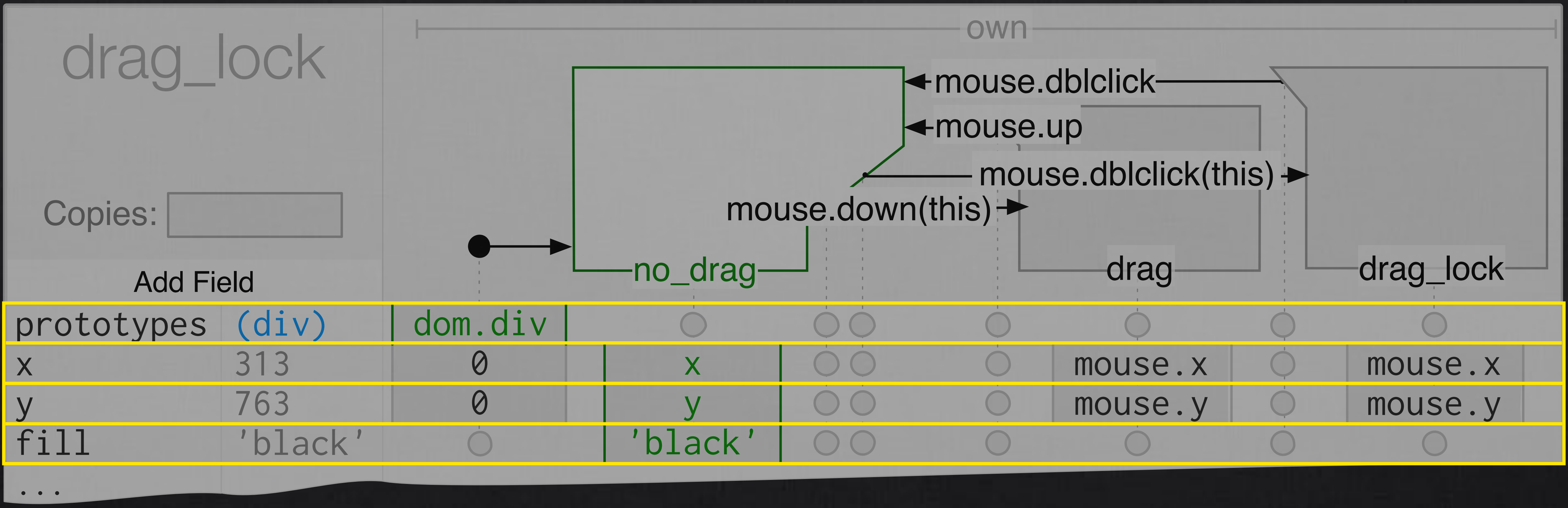


drag_lock

Copies:

Add Field

prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									



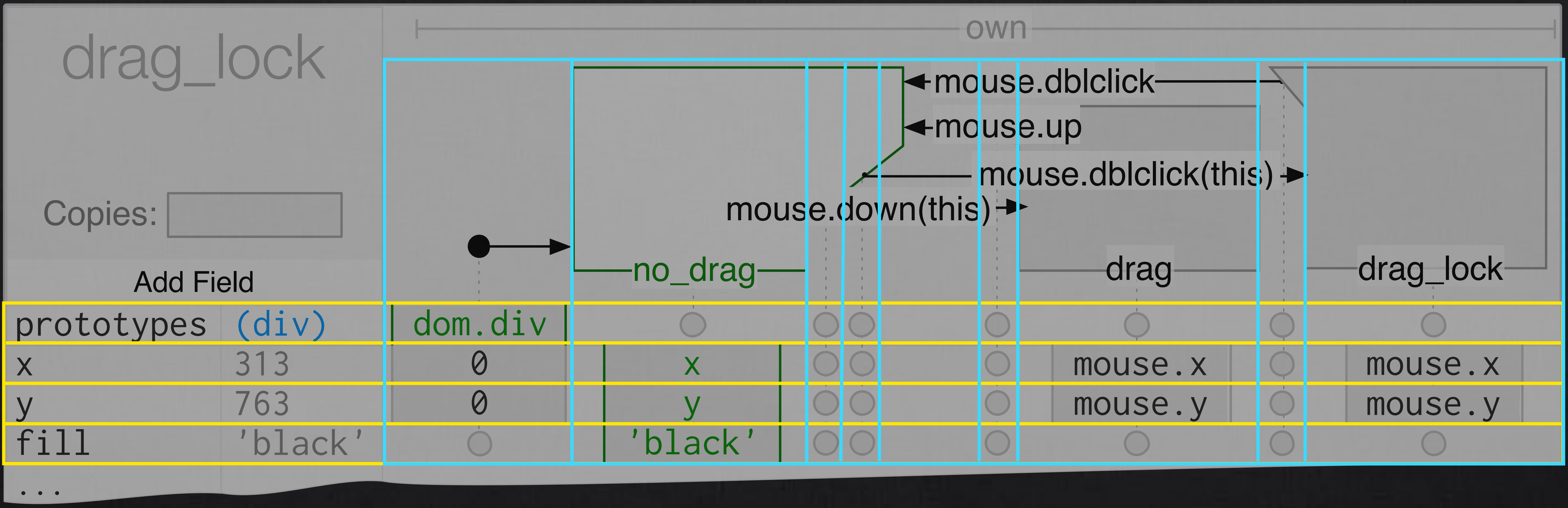
drag_lock

Copies:

Add Field

prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○

...



drag_lock

Copies:

Add Field

own

mouse.dblclick

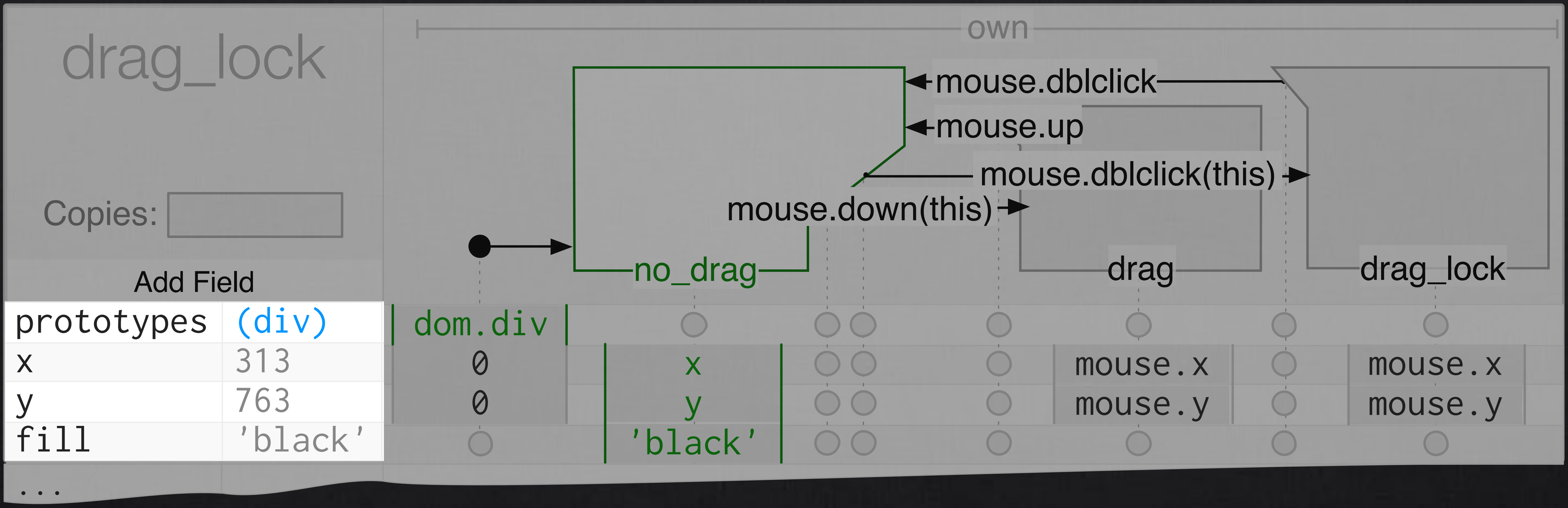
mouse.up

mouse.dblclick(this)

mouse.down(this)

			no_drag				drag		drag_lock
prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○

...



drag_lock

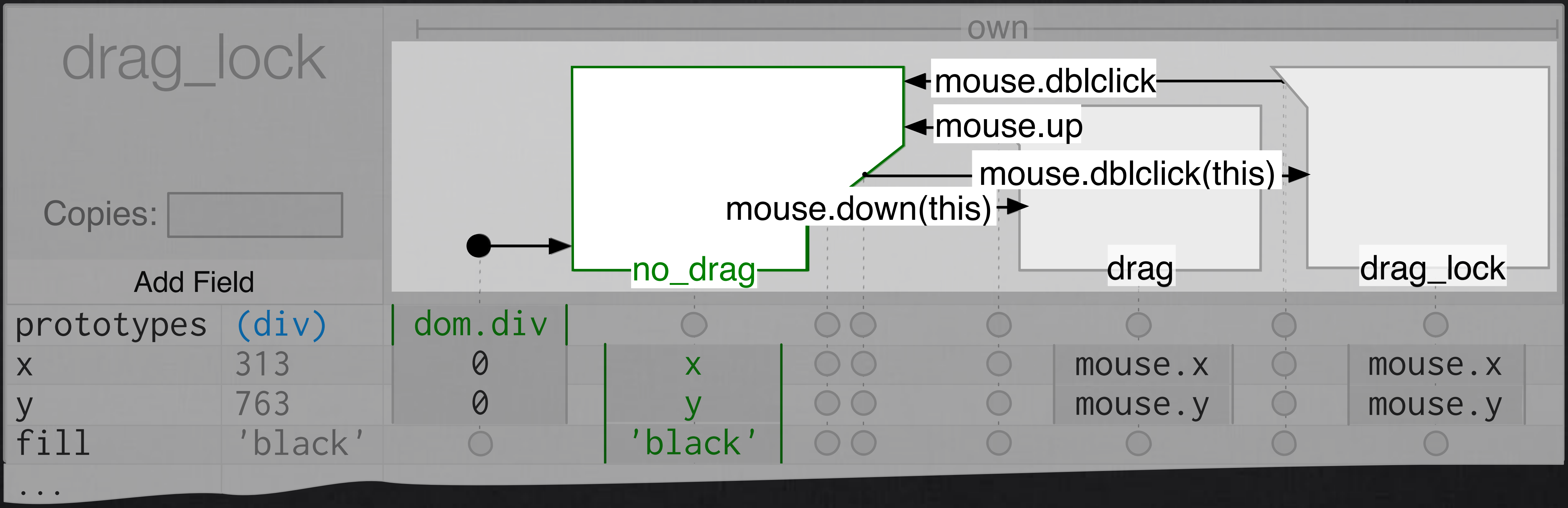
Copies:

Add Field

prototypes	(div)
x	313
y	763
fill	'black'
...	

fields

state machine

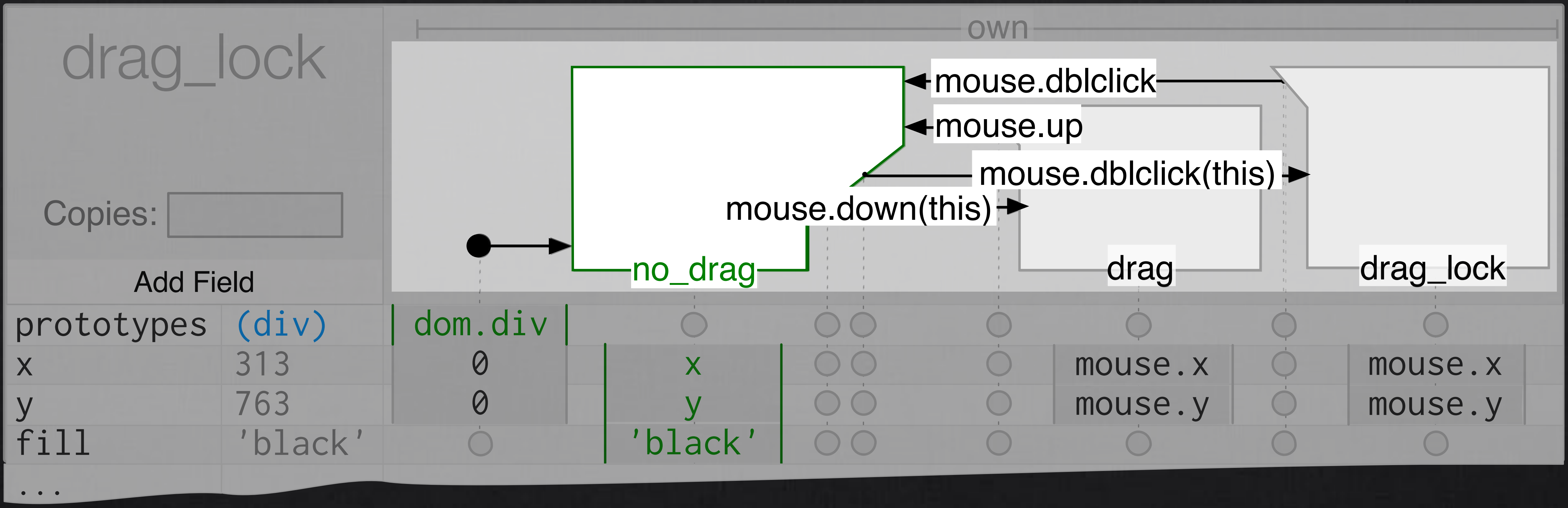


prototypes	(div)	dom.div	○	○	○
x	313	0	x	mouse.x	mouse.x
y	763	0	y	mouse.y	mouse.y
fill	'black'	○	'black'	○	○
...					

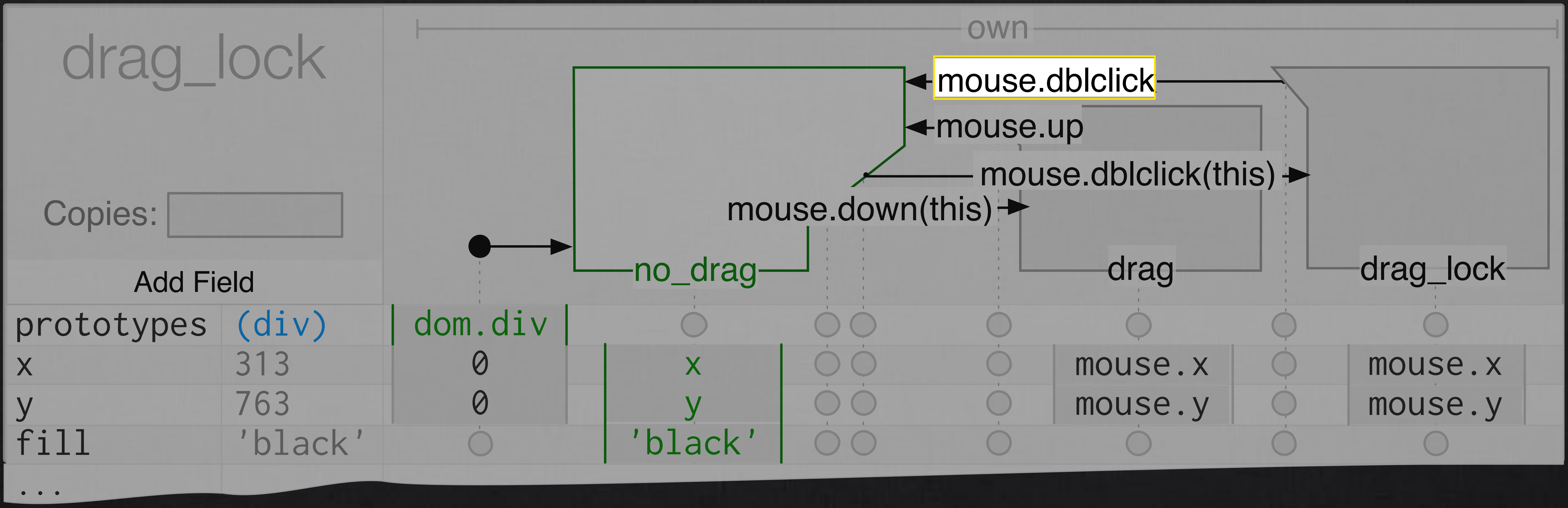
```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
    }
    isDragLocked = !isDragLocked;
});
```

```
  var isDragLocked = false,
    mm_listener  = function(mm_ev) {
      draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
+     if(!isDragLocked) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
+     }
    };
  draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
+   if(!isDragLocked) {
      addEventListener("mousemove", mm_listener);
      addEventListener("mouseup",   mu_listener);
+   }
  }).dblclick(function(md_event) {
    if(!isDragLocked) {
- ~~     removeEventListener("mousemove", mm_listener);~~
- ~~  } else {~~
         addEventListener("mousemove", mm_listener);
      isDragLocked = !isDragLocked;
    }
+ }).click(function(c_event) {
+   if(isDragLocked) {
+     removeEventListener("mousemove", mm_listener);
+     isDragLocked = !isDragLocked;
+   }
+ });
```

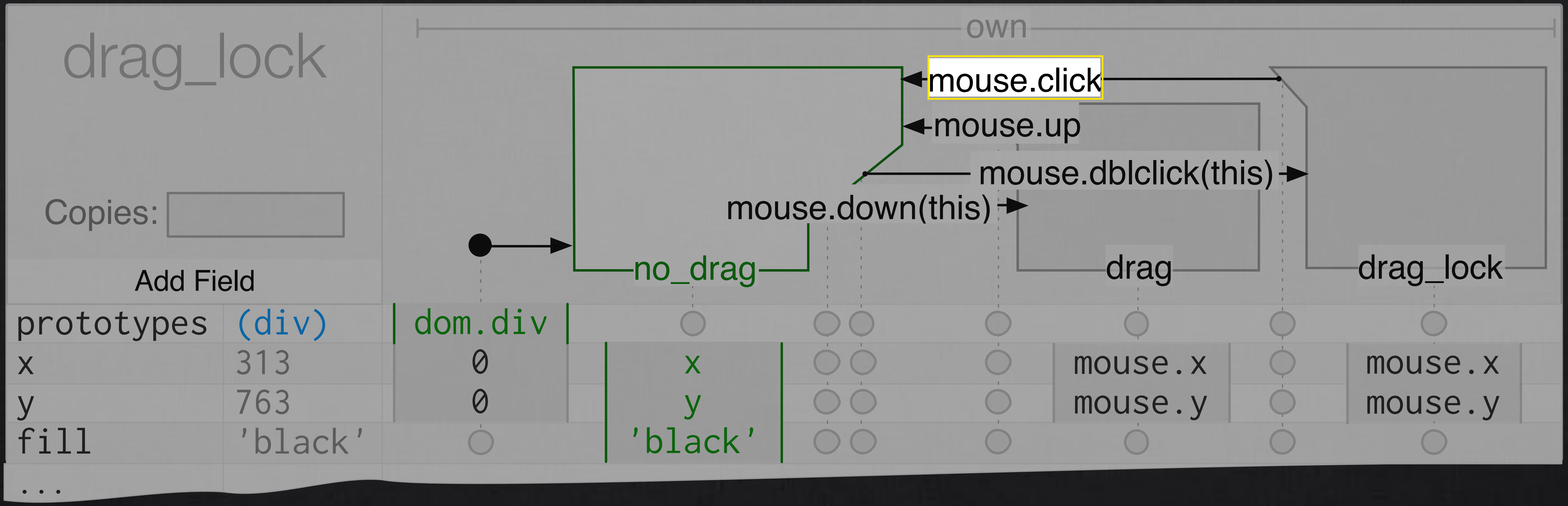
state machine



prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									



prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									

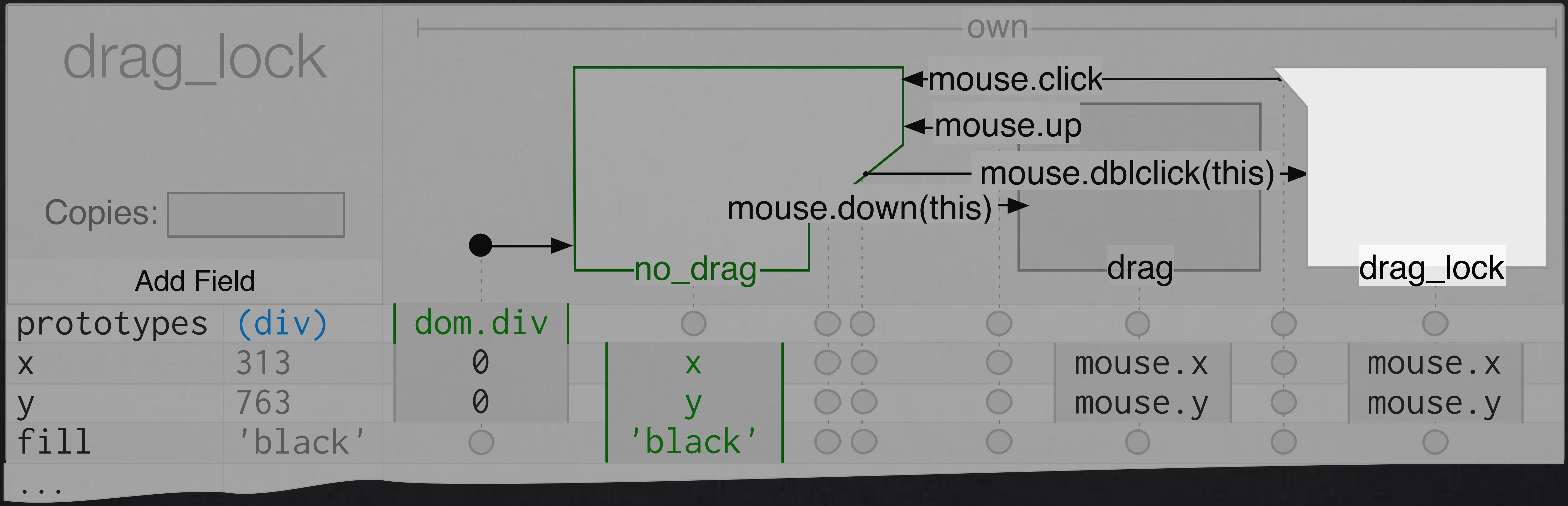


drag_lock

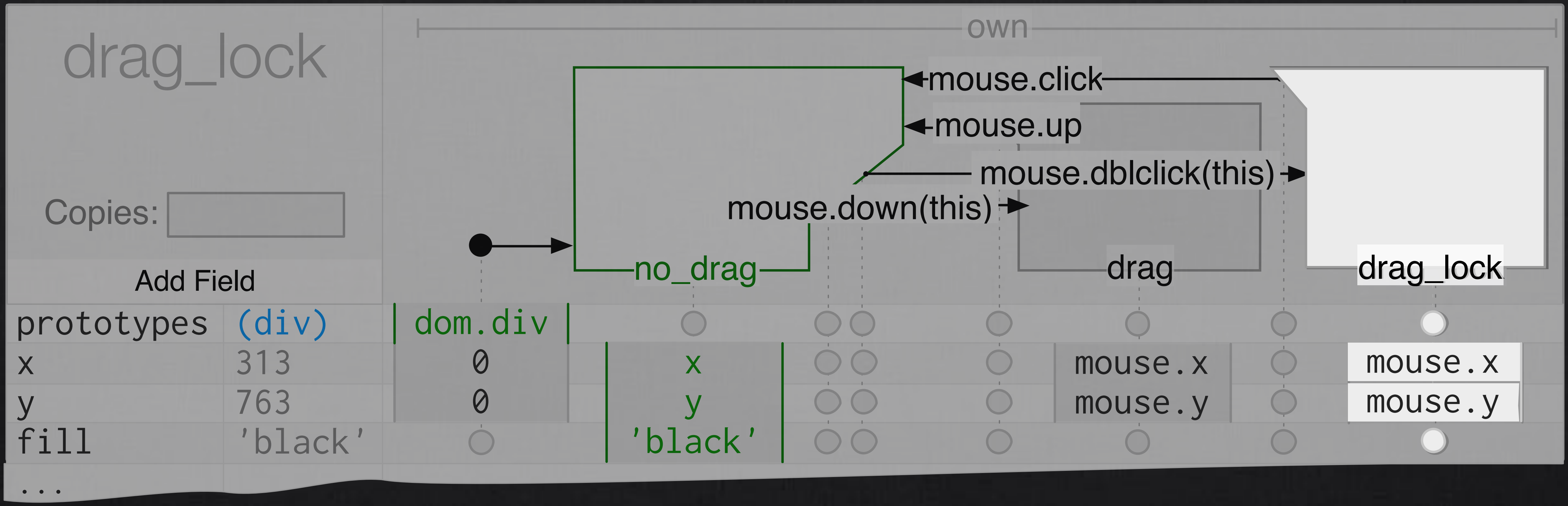
Copies:

Add Field

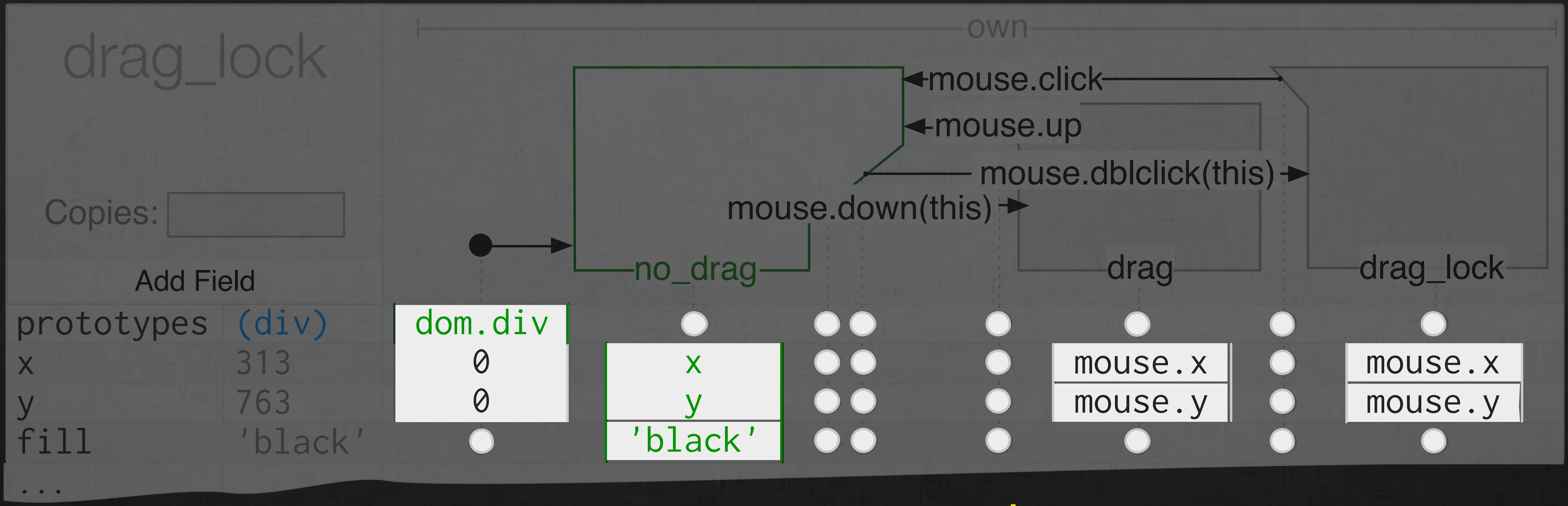
prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									



prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									

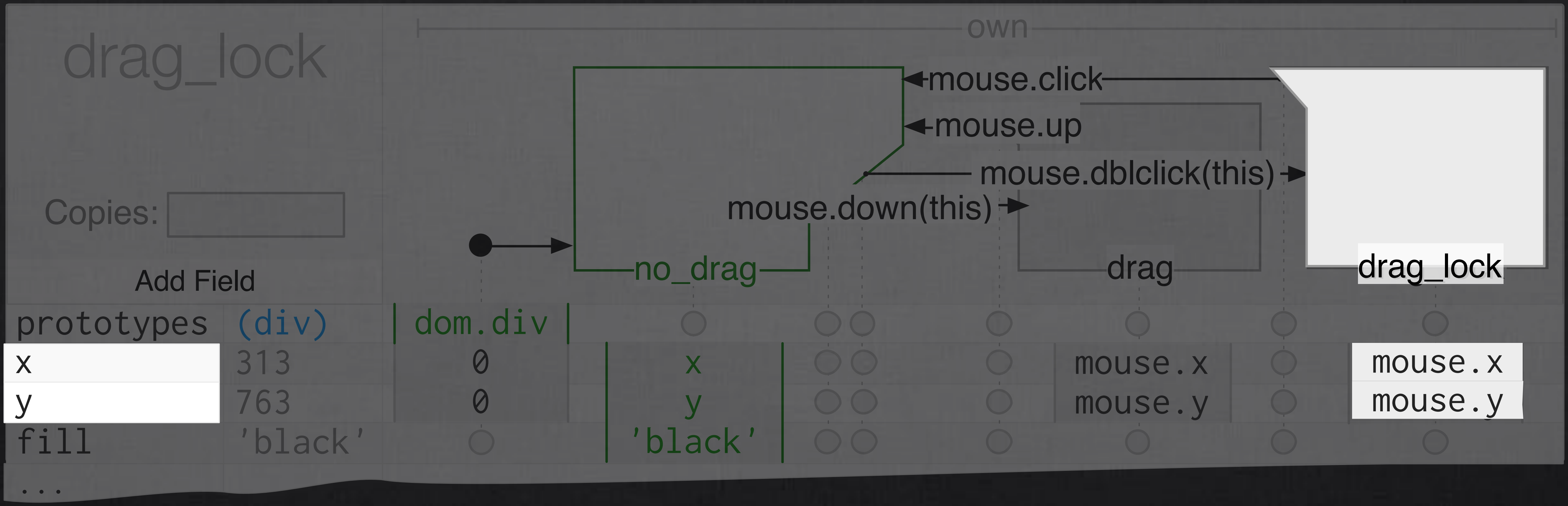


prototypes	(div)	dom.div	○	○	○
x	313	0	x	mouse.x	mouse.x
y	763	0	y	mouse.y	mouse.y
fill	'black'	○	'black'	○	○
...					



			no_drag	drag	drag_lock
prototypes	(div)	dom.div			
x	313	0	x	mouse.x	mouse.x
y	763	0	y	mouse.y	mouse.y
fill	'black'		'black'		
...					

constraints



drag_lock

Copies:

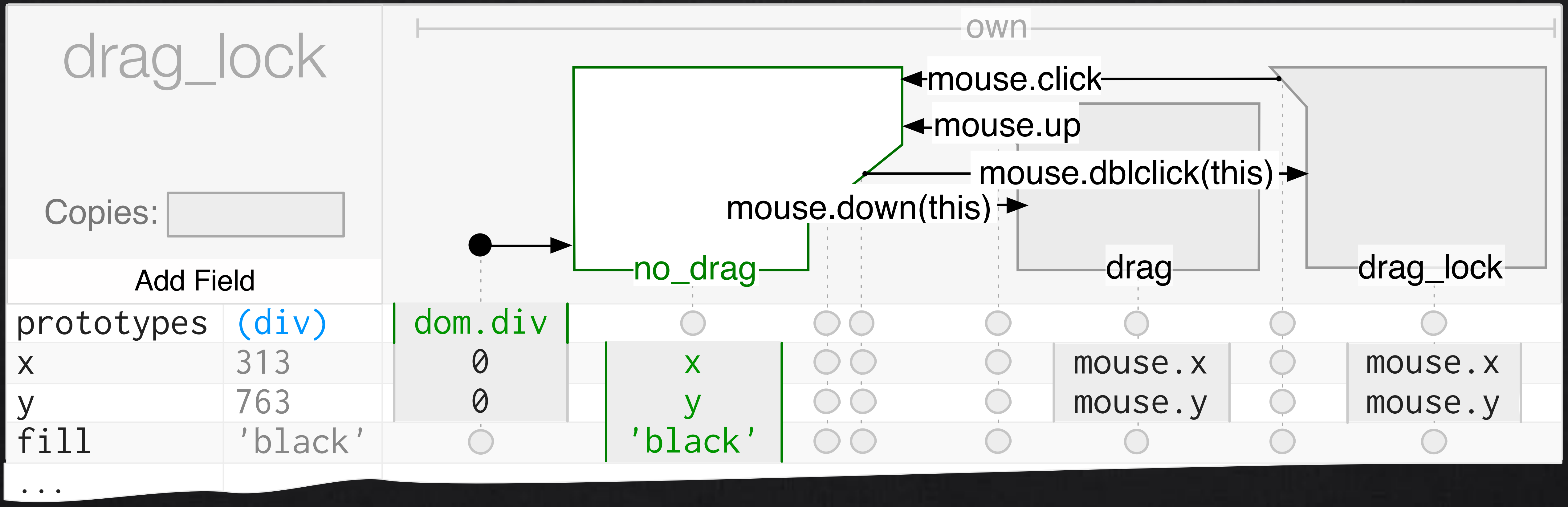
Add Field

prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									

```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
    }
    isDragLocked = !isDragLocked;
});
```

```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
           draggable.attr({ fill: “black" });
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
     draggable.attr({ fill: “blue" });
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
           draggable.attr({ fill: “navy" });
    }
    isDragLocked = !isDragLocked;
});
draggable.attr({ fill: “black" });
setInitialValues(draggable);
```

```
var isDragLocked = false,
    mm_listener  = function(mm_event) {
        draggable.attr({ x: mm_ev.x, y: mm_ev.y });
    },
    mu_listener  = function(mu_event) {
        removeEventListener("mousemove", mm_listener);
        removeEventListener("mouseup",   mu_listener);
           draggable.attr({ fill: “black" });
    };

draggable.mousedown(function(md_ev) {
    draggable.attr({ x: md_ev.x, y: md_ev.y });
    addEventListener("mousemove", mm_listener);
    addEventListener("mouseup",   mu_listener);
     draggable.attr({ fill: “blue" });
}).dblclick(function(md_event) {
    if(isDragLocked) {
        removeEventListener("mousemove", mm_listener);
    } else {
        addEventListener   ("mousemove", mm_listener);
           draggable.attr({ fill: “navy" });
    }
    isDragLocked = !isDragLocked;
});
draggable.attr({ fill: “black" });
setInitialValues(draggable);
```

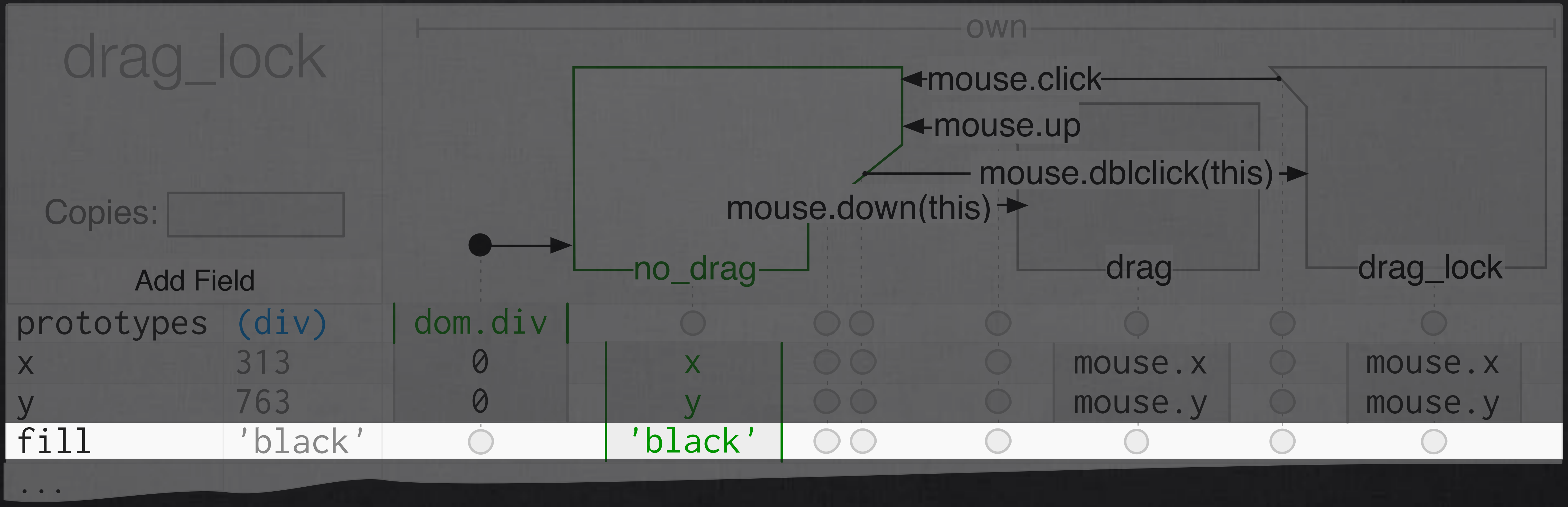


drag_lock

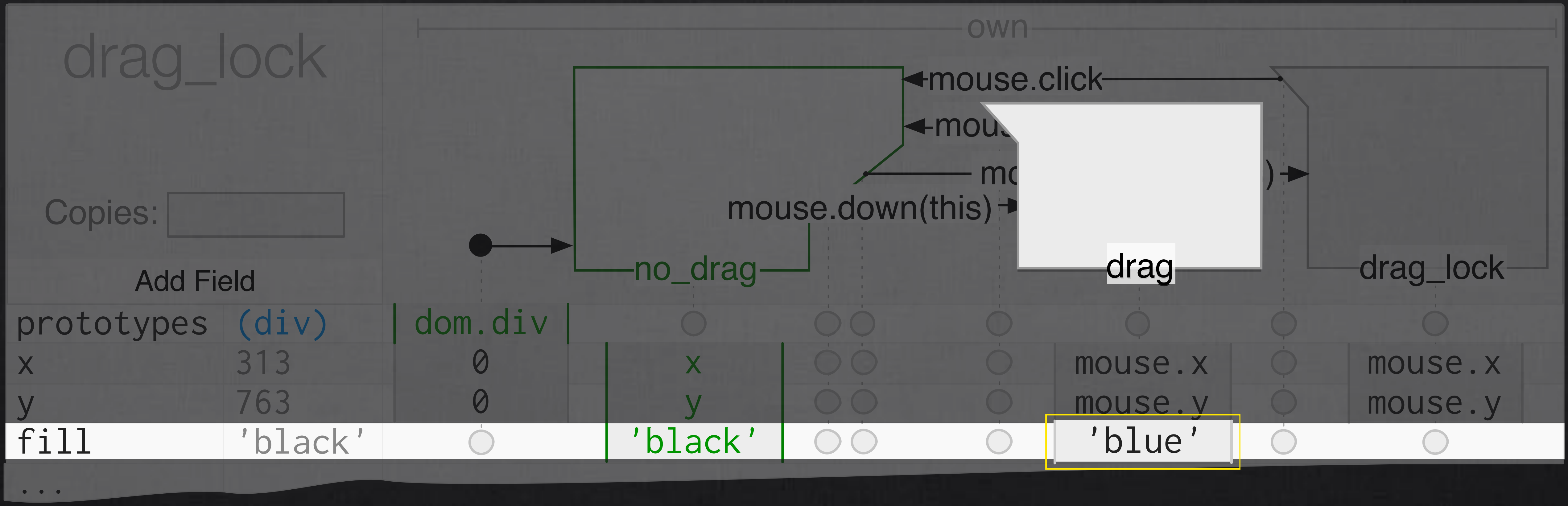
Copies:

Add Field

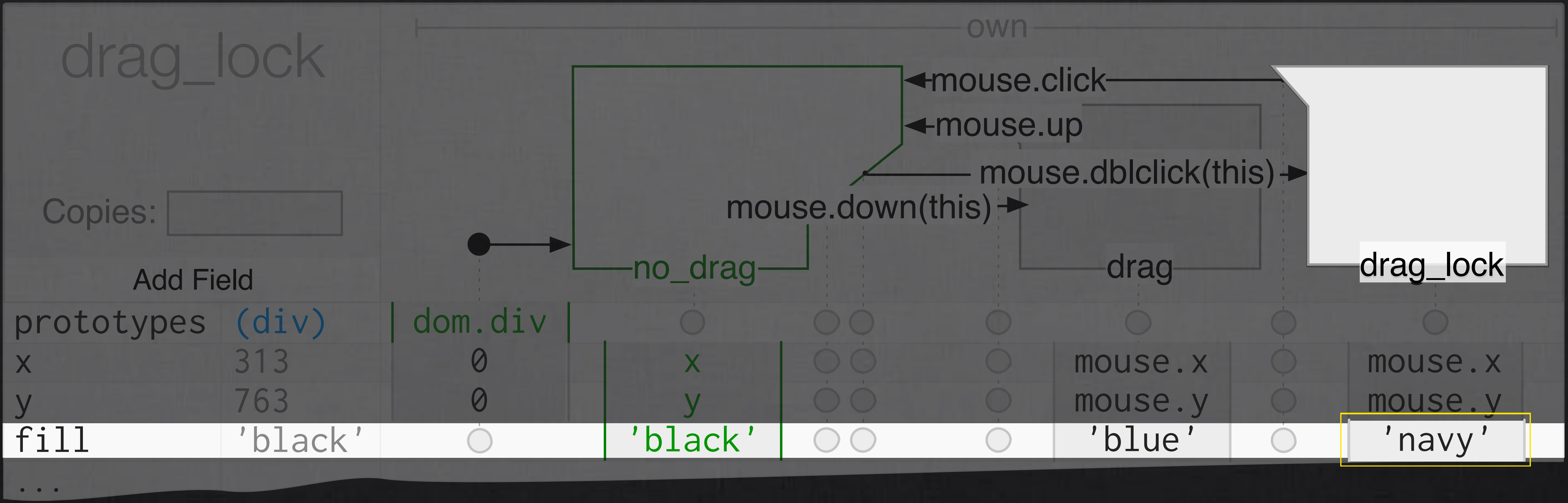
prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									



prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	○	○	○
...									



prototypes	(div)	dom.div	○	○ ○	○	○	○	○
x	313	0	x	○ ○	○	mouse.x	○	mouse.x
y	763	0	y	○ ○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○ ○	○	'blue'	○	○
...								

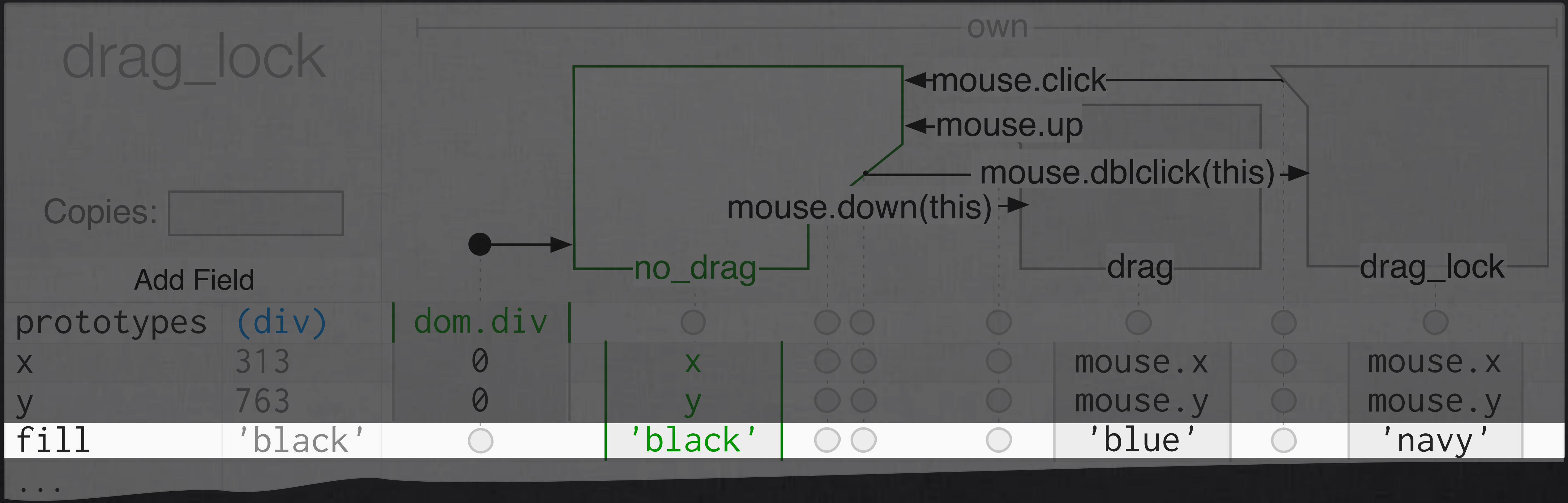


drag_lock

Copies:

Add Field

			no_drag				drag		drag_lock
prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	'blue'	○	'navy'
...									

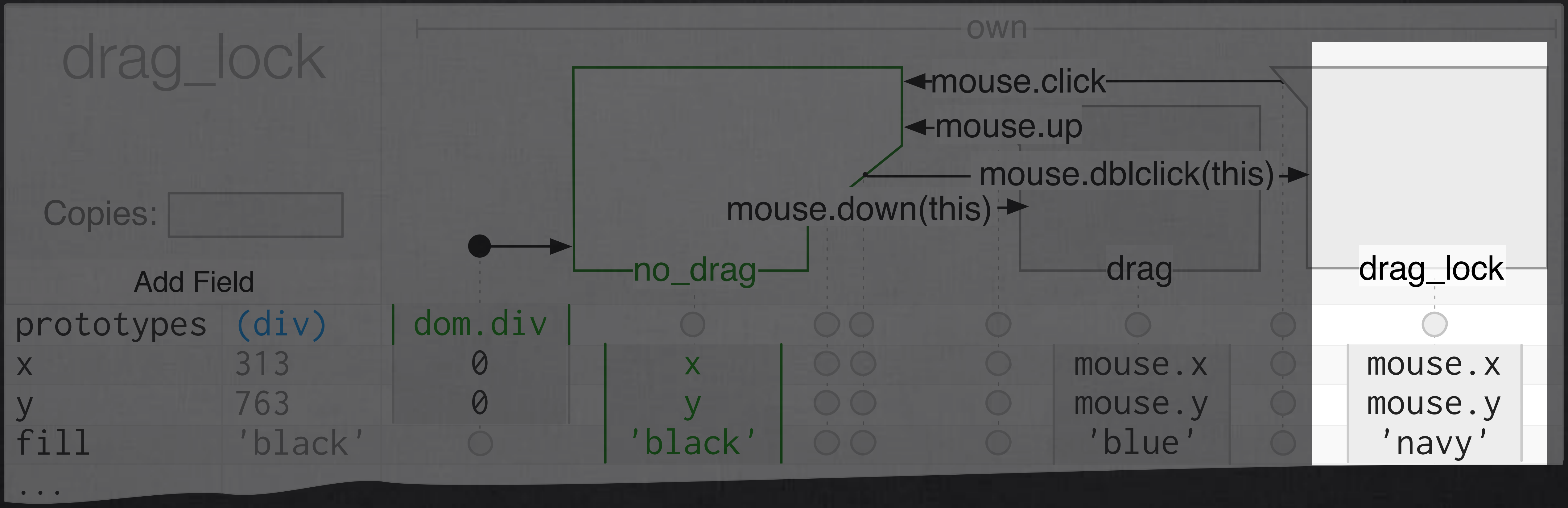


drag_lock

Copies:

Add Field

prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	'blue'	○	'navy'
...									

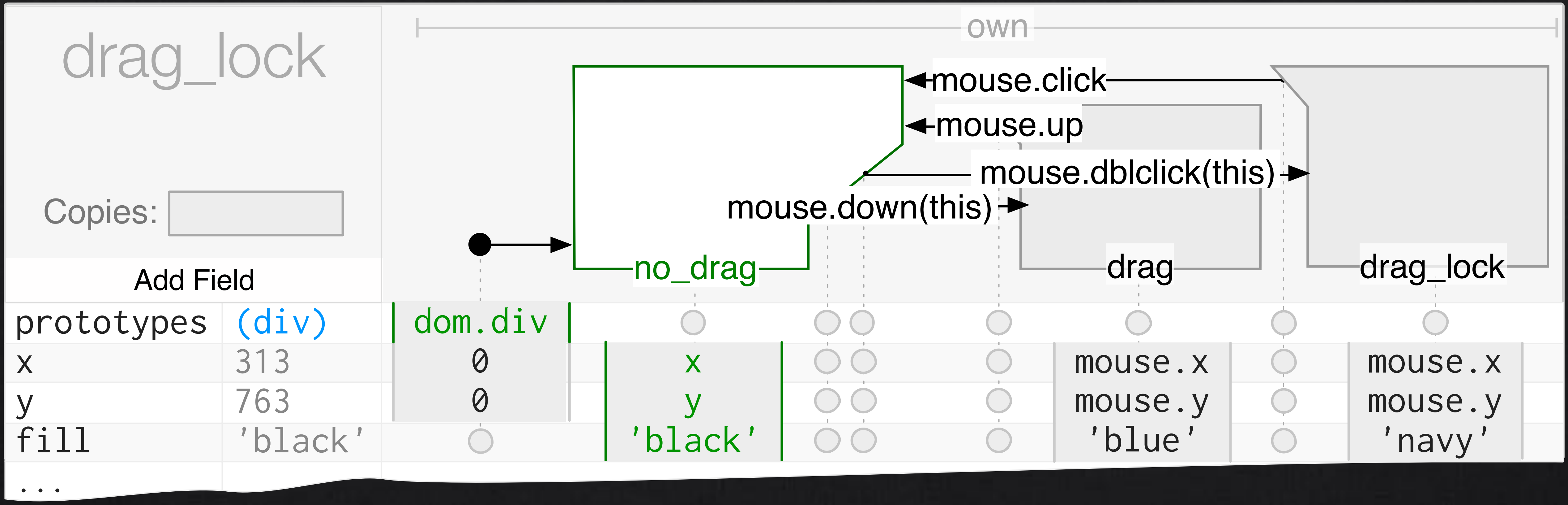


drag_lock

Copies:

Add Field

			no_drag	drag	drag_lock
prototypes	(div)	dom.div	○	○	○
x	313	0	x	mouse.x	mouse.x
y	763	0	y	mouse.y	mouse.y
fill	'black'	○	'black'	'blue'	'navy'
...					



drag_lock

Copies:

Add Field

prototypes	(div)	dom.div	○	○	○	○	○	○	○
x	313	0	x	○	○	○	mouse.x	○	mouse.x
y	763	0	y	○	○	○	mouse.y	○	mouse.y
fill	'black'	○	'black'	○	○	○	'blue'	○	'navy'
...									

InterState Contributions

Computational Model

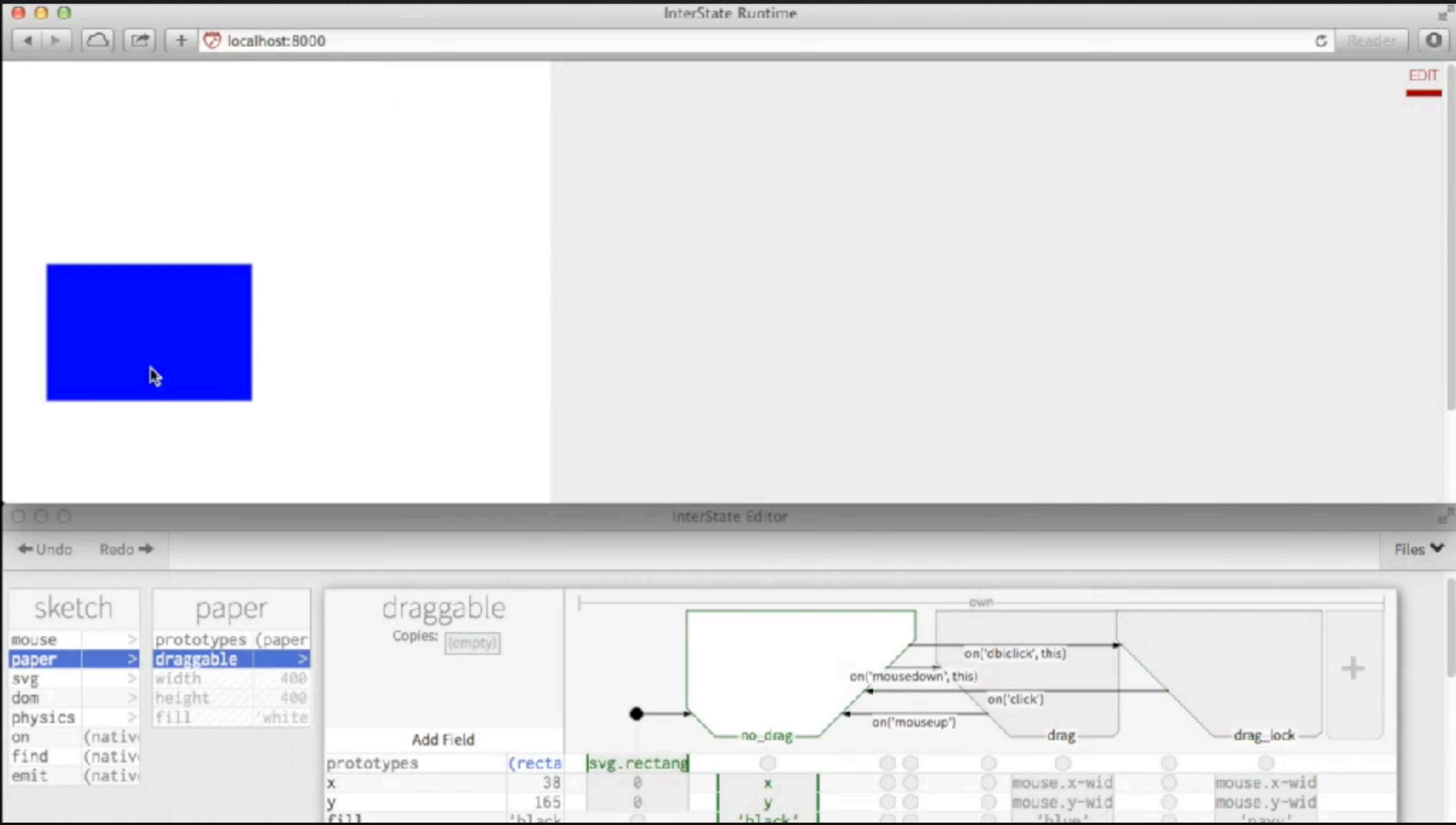
- clear & concise
- states & constraints

Visual Notation

- suits computational model
- states & constraints in grid

Live Editor

- current states/values
- immediate effects



sketch

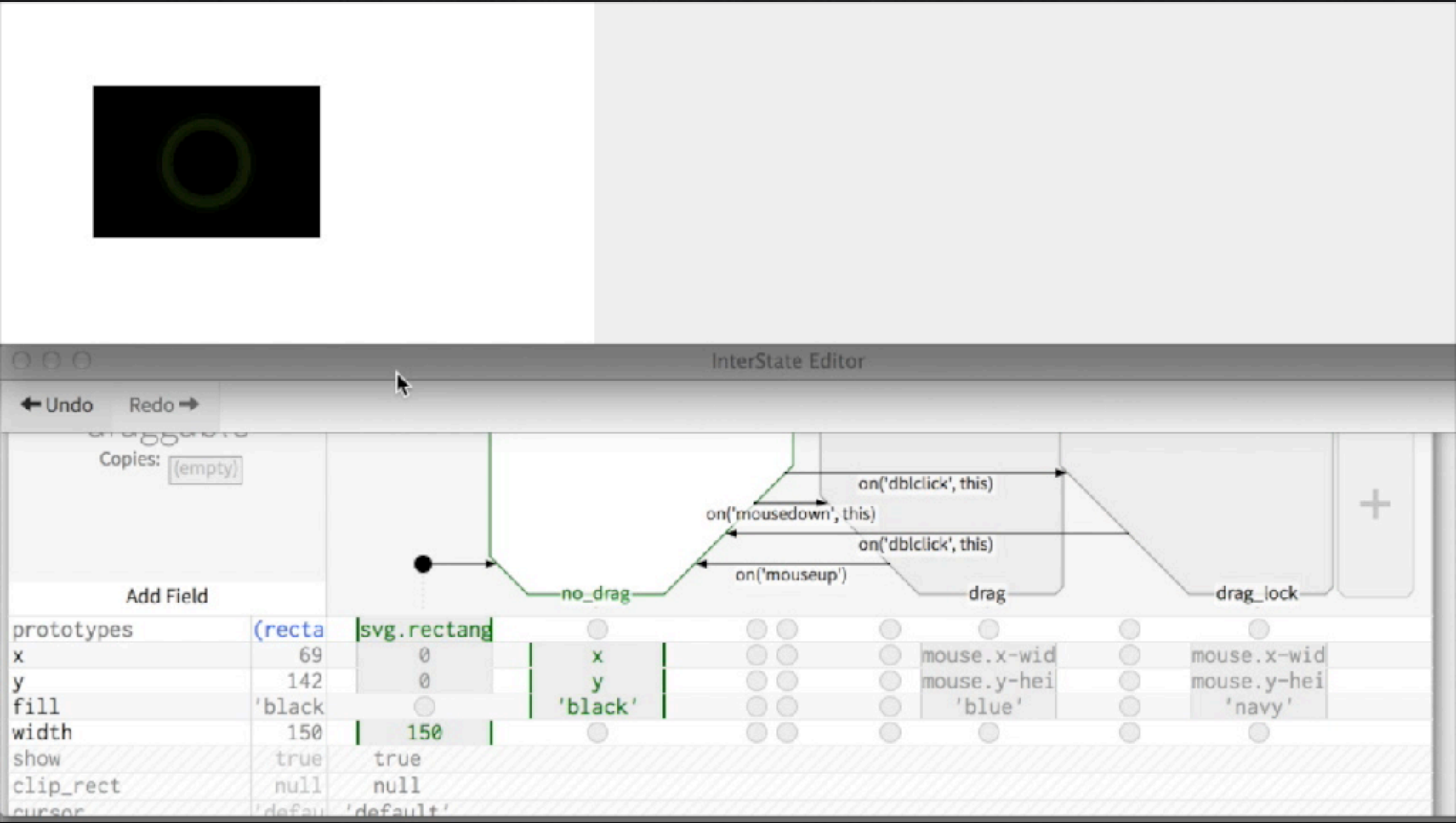
mouse	>
paper	>
svg	>
dom	>
physics	>
on	(nativ
find	(nativ
emit	(nativ

paper

prototypes	(paper
draggable	>
width	400
height	400
fill	'white

draggable

			no_drag	drag	drag_lock
prototypes	(recta	svg.rectang			
x	38	0	x	mouse.x-wid	mouse.x-wid
y	165	0	y	mouse.y-wid	mouse.y-wid
fill	'black		'black'	'blue'	'navy'



			no_drag				drag		drag_lock
prototypes	(recta	svg.rectang	○	○	○	○	○	○	○
x	69	0	x	○	○	○	mouse.x-wid	○	mouse.x-wid
y	142	0	y	○	○	○	mouse.y-hei	○	mouse.y-hei
fill	'black	○	'black'	○	○	○	'blue'	○	'navy'
width	150	150	○	○	○	○	○	○	○
show	true	true							
clip_rect	null	null							
cursor	'defau	'default'							

InterState Contributions

Computational Model

- clear & concise
- states & constraints

Visual Notation

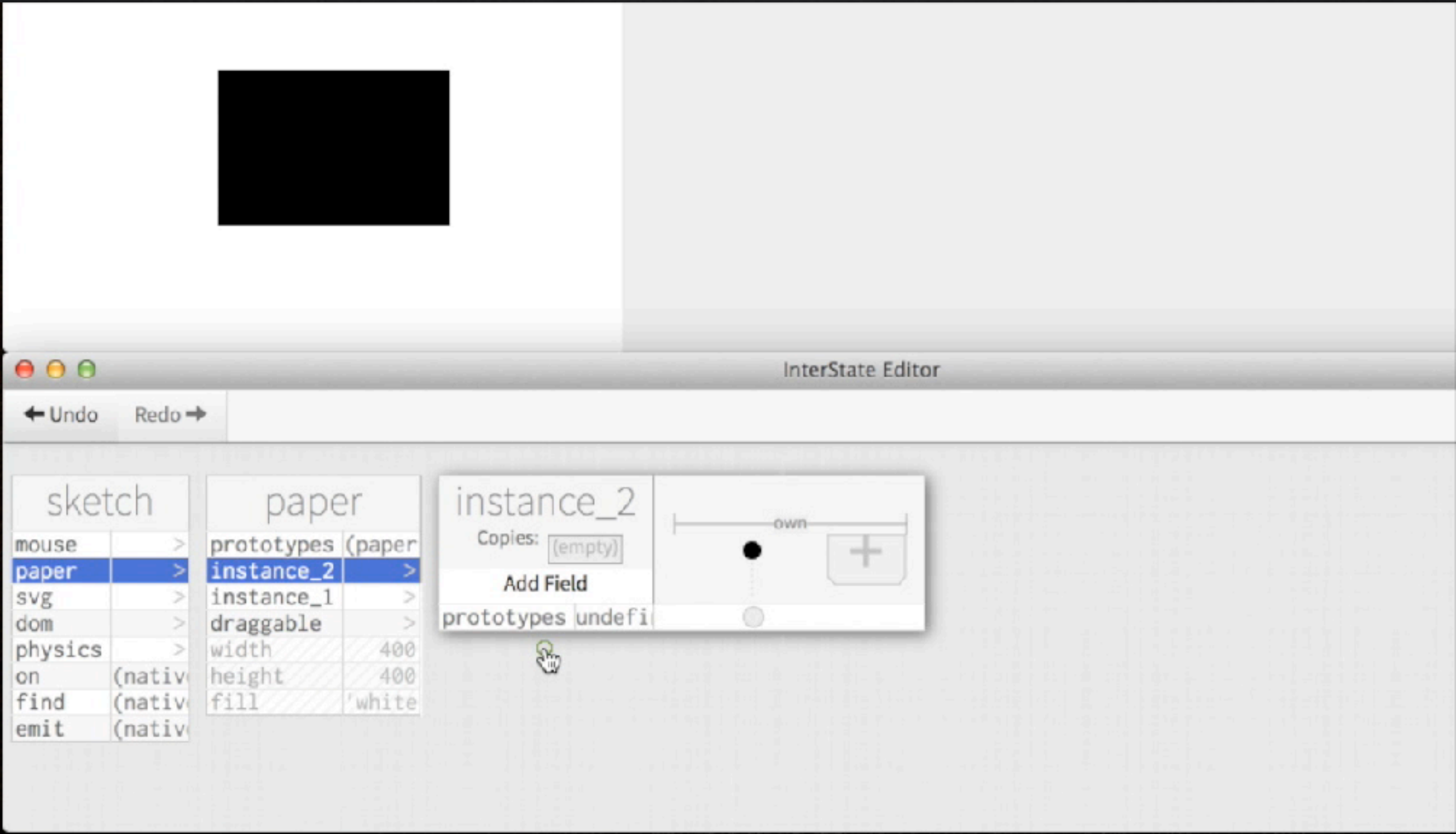
- suits computational model
- states & constraints in grid

Live Editor

- current states/values
- immediate effects

Behavior Re-use

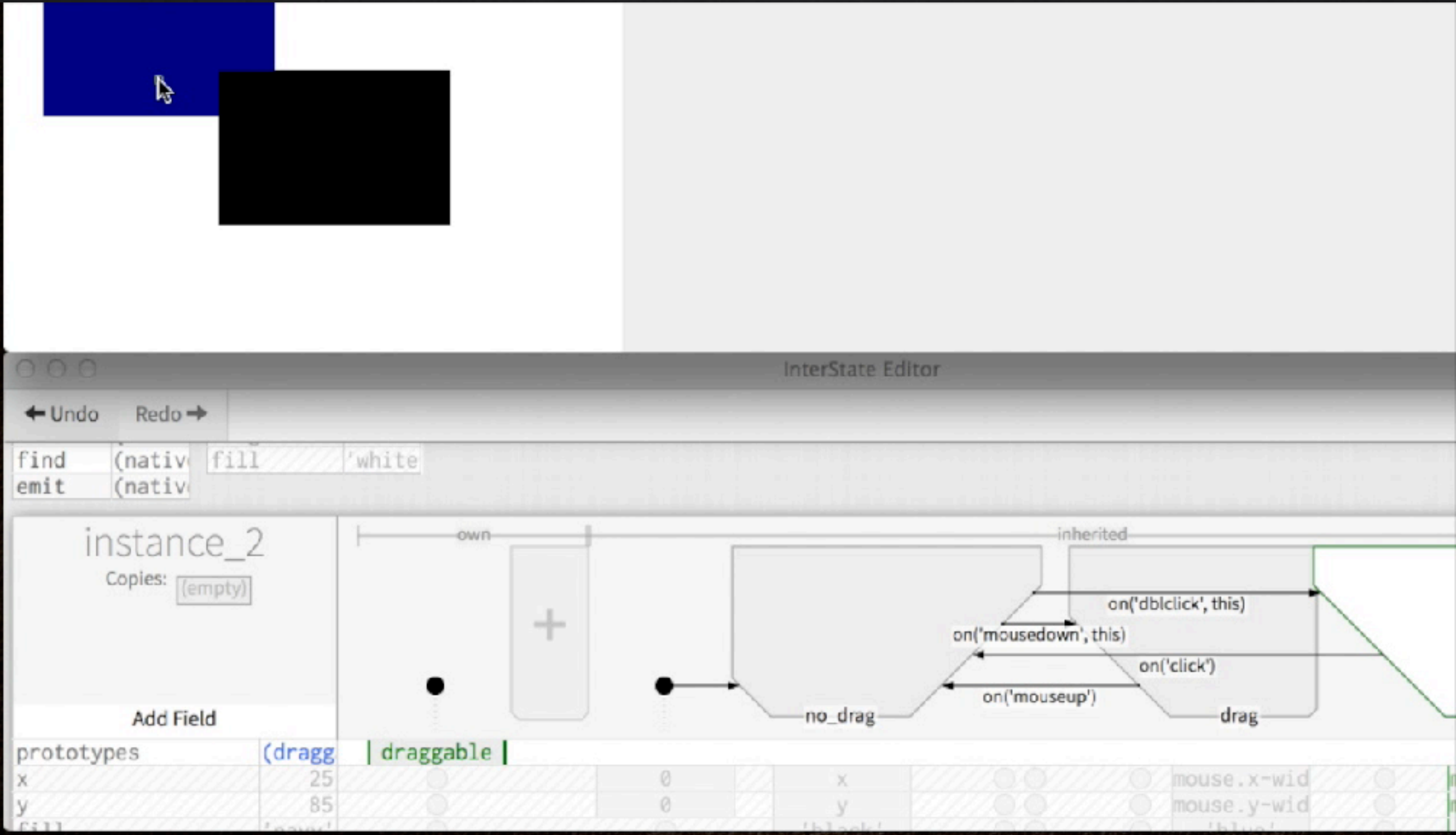
- inheritance mechanism
- template re-use



sketch	
mouse	>
paper	>
svg	>
dom	>
physics	>
on	(nativ
find	(nativ
emit	(nativ

paper	
prototypes	(paper
instance_2	>
instance_1	>
draggable	>
width	400
height	400
fill	'white

instance_2	
prototypes	undefi

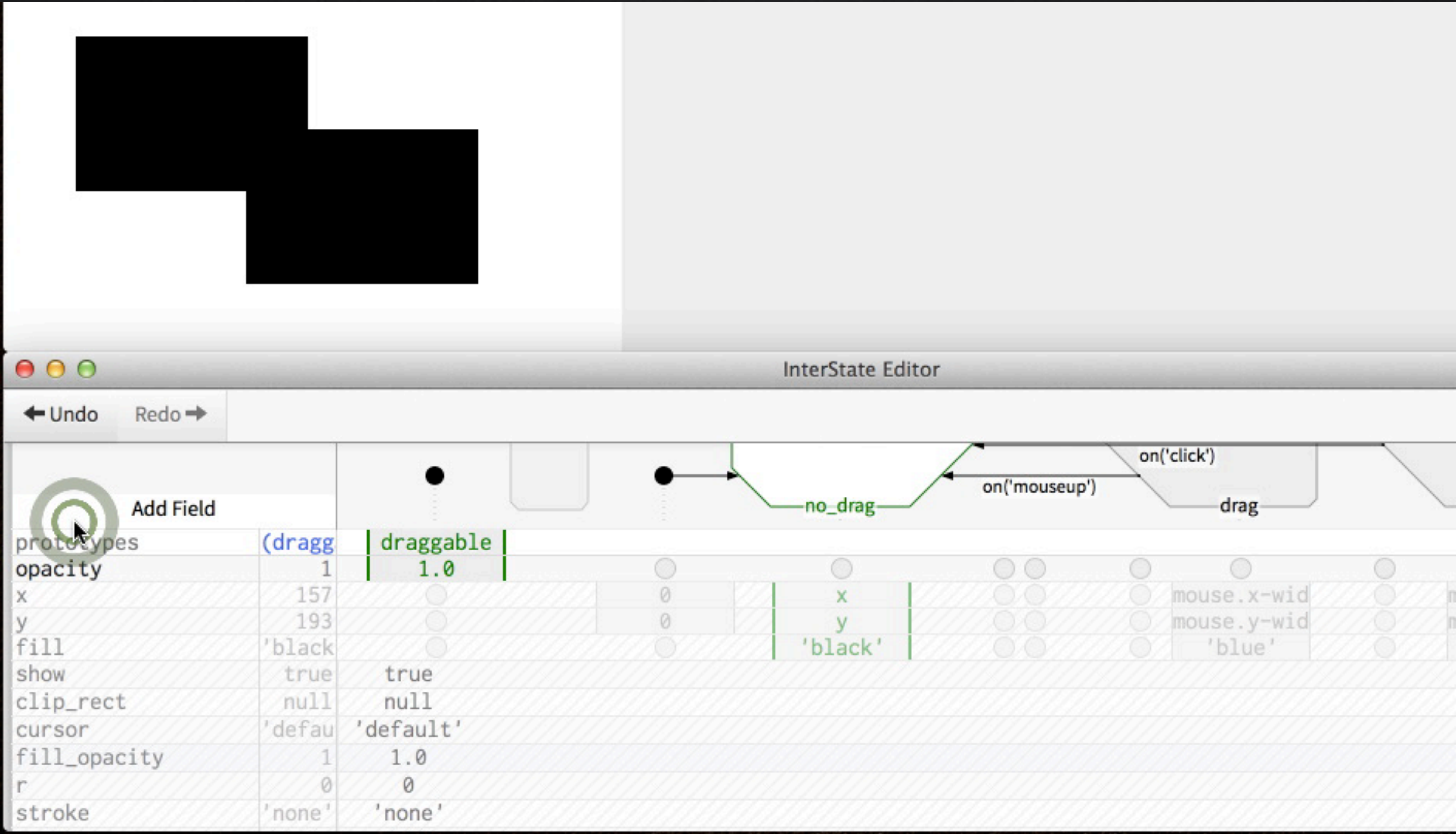


find (nativ
emit (nativ
fill 'white

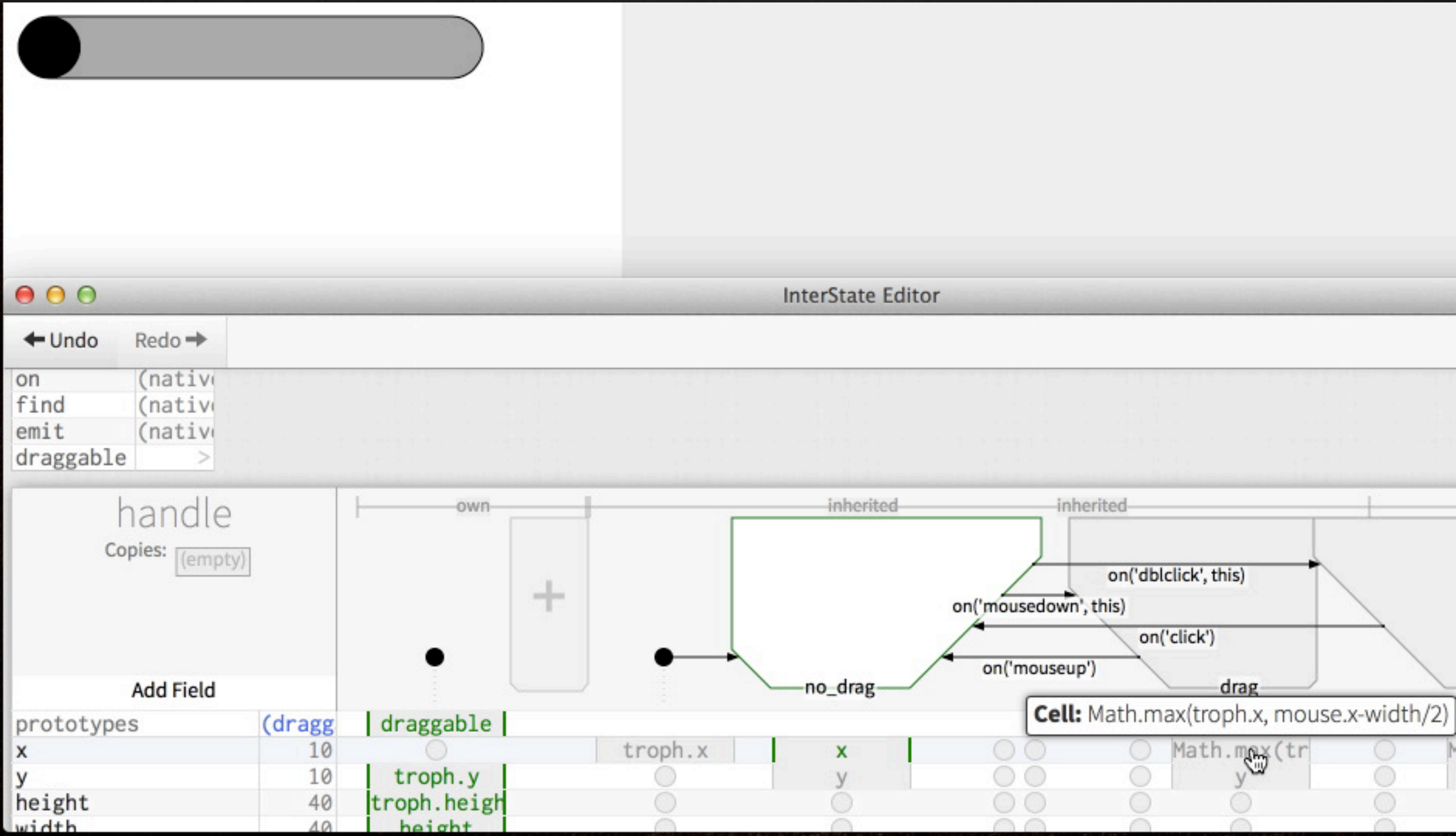
instance_2
Copies: (empty)

prototypes (dragg | draggable |

x	25	0	x	mouse.x-wid
y	85	0	y	mouse.y-wid



				no_drag			drag	
prototypes	(dragg	draggable						
opacity	1	1.0	○	○	○ ○	○	○	○
x	157	○	0	x	○ ○	○	mouse.x-wid	○
y	193	○	0	y	○ ○	○	mouse.y-wid	○
fill	'black	○	○	'black'	○ ○	○	'blue'	○
show	true	true						
clip_rect	null	null						
cursor	'defau	'default'						
fill_opacity	1	1.0						
r	0	0						
stroke	'none'	'none'						



on	(nativ
find	(nativ
emit	(nativ
draggable	>

Cell: Math.max(troph.x, mouse.x-width/2)

prototypes	(dragg	draggable			
x	10	○	troph.x	x	Math.max(tr
y	10	troph.y	○	y	y
height	40	troph.heigh	○	○	○
width	40	height	○	○	○

InterState Contributions

Computational Model

- clear & concise
- states & constraints

Visual Notation

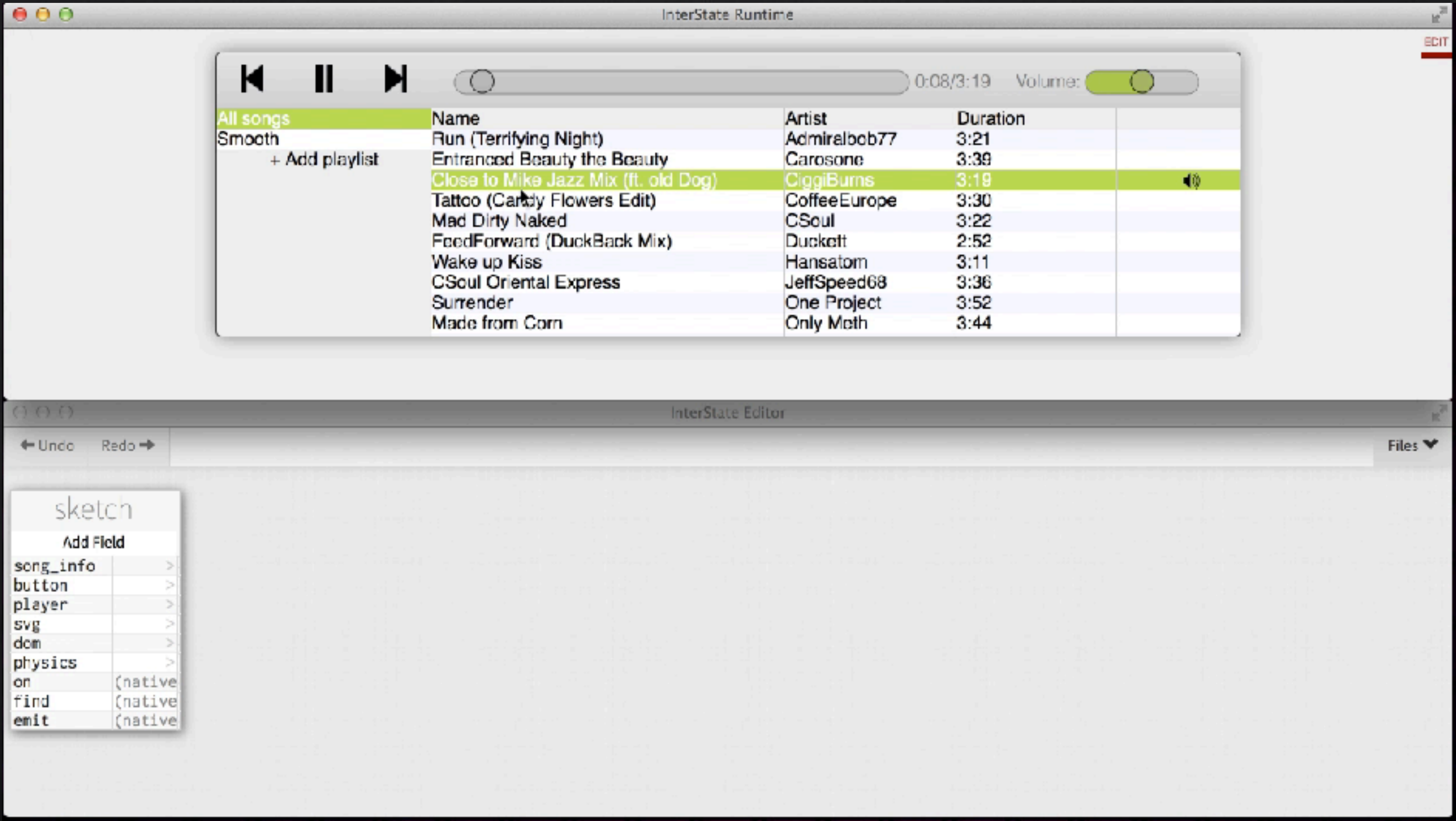
- suits computational model
- states & constraints in grid

Live Editor

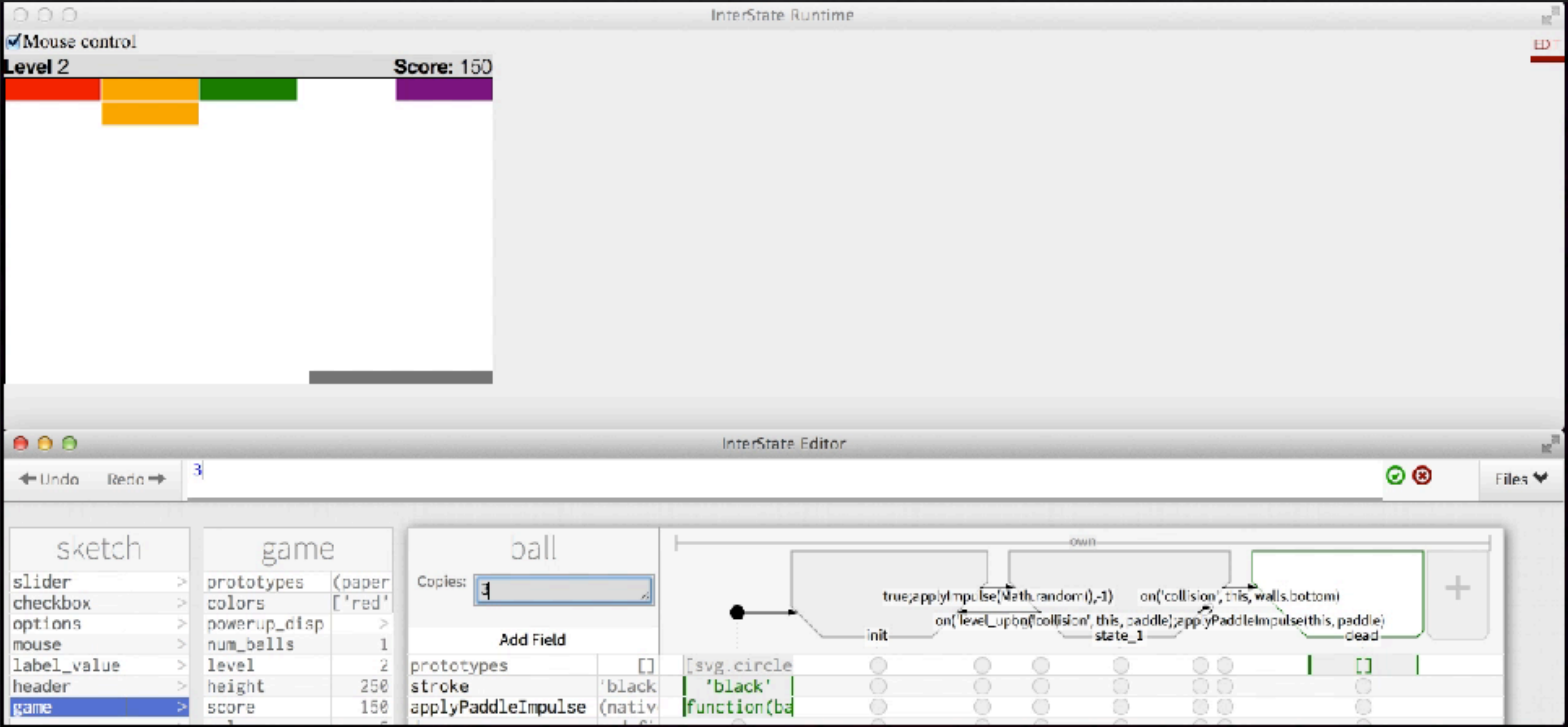
- current states/values
- immediate effects

Behavior Re-use

- inheritance mechanism
- template re-use



Name	Artist	Duration
Run (Terrifying Night)	Admiralbob77	3:21
Entranced Beauty the Beauty	Carosone	3:39
Close to Mike Jazz Mix (ft. old Dog)	CiggiBurns	3:19
Tattoo (Candy Flowers Edit)	CoffeeEurope	3:30
Mad Dirty Naked	CSoul	3:22
FeedForward (DuckBack Mix)	Duckett	2:52
Wake up Kiss	Hansatom	3:11
CSoul Oriental Express	JeffSpeed68	3:36
Surrender	One Project	3:52
Made from Corn	Only Meth	3:44



3

sketch	
slider	>
checkbox	>
options	>
mouse	>
label_value	>
header	>
game	>

game	
prototypes	(paper
colors	['red'
powerup_disp	>
num_balls	1
level	2
height	250
score	150

Copies: 3

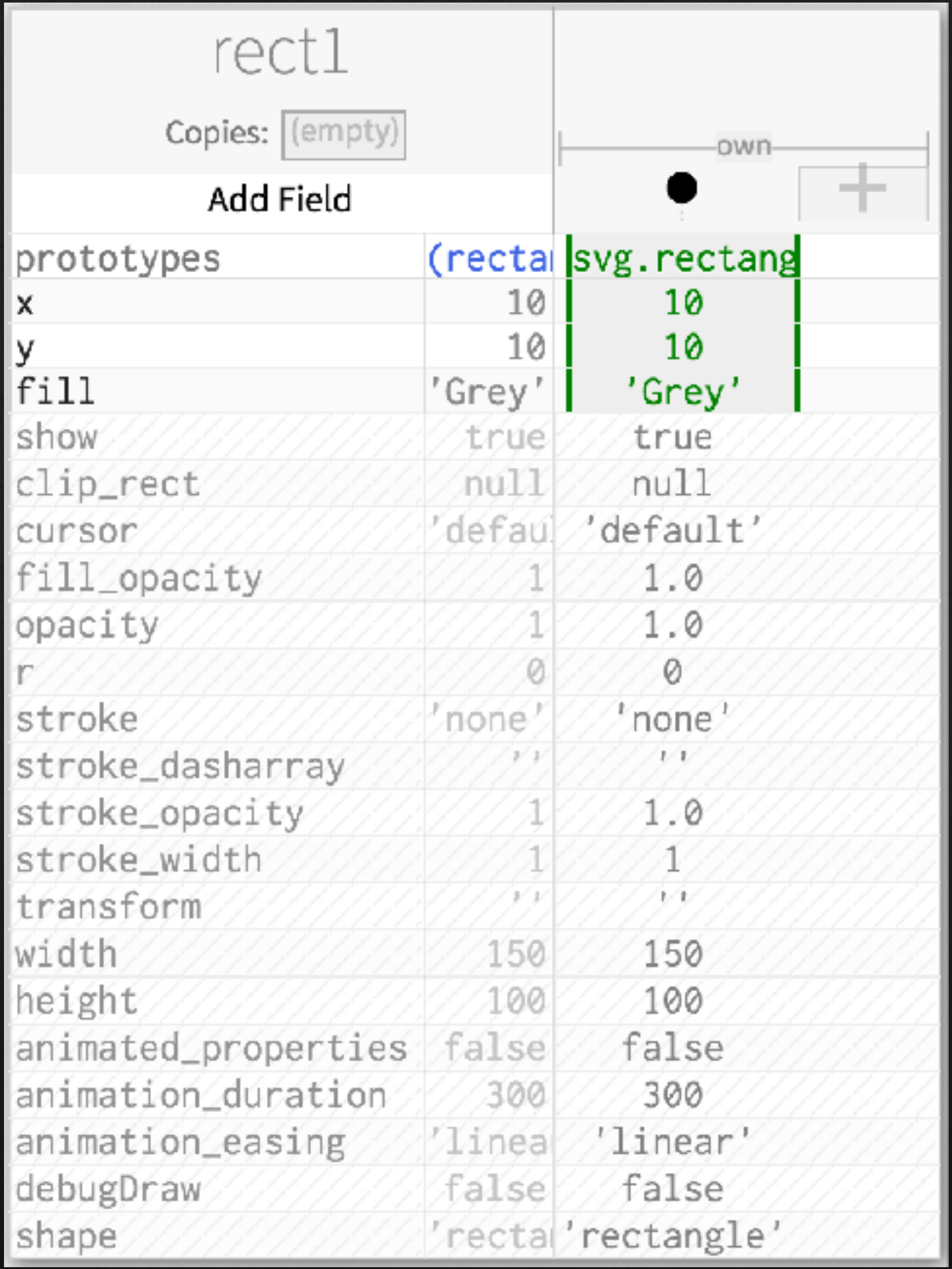
ball		
prototypes	[]	[svg.circle
stroke	'black	'black'
applyPaddleImpulse	(nativ	function(ba

true;applyImpulse(Math.random(),-1)

on('level_up')

on('collision', this, paddle);applyPaddleImpulse(this, paddle)

on('collision', this, walls.bottom)

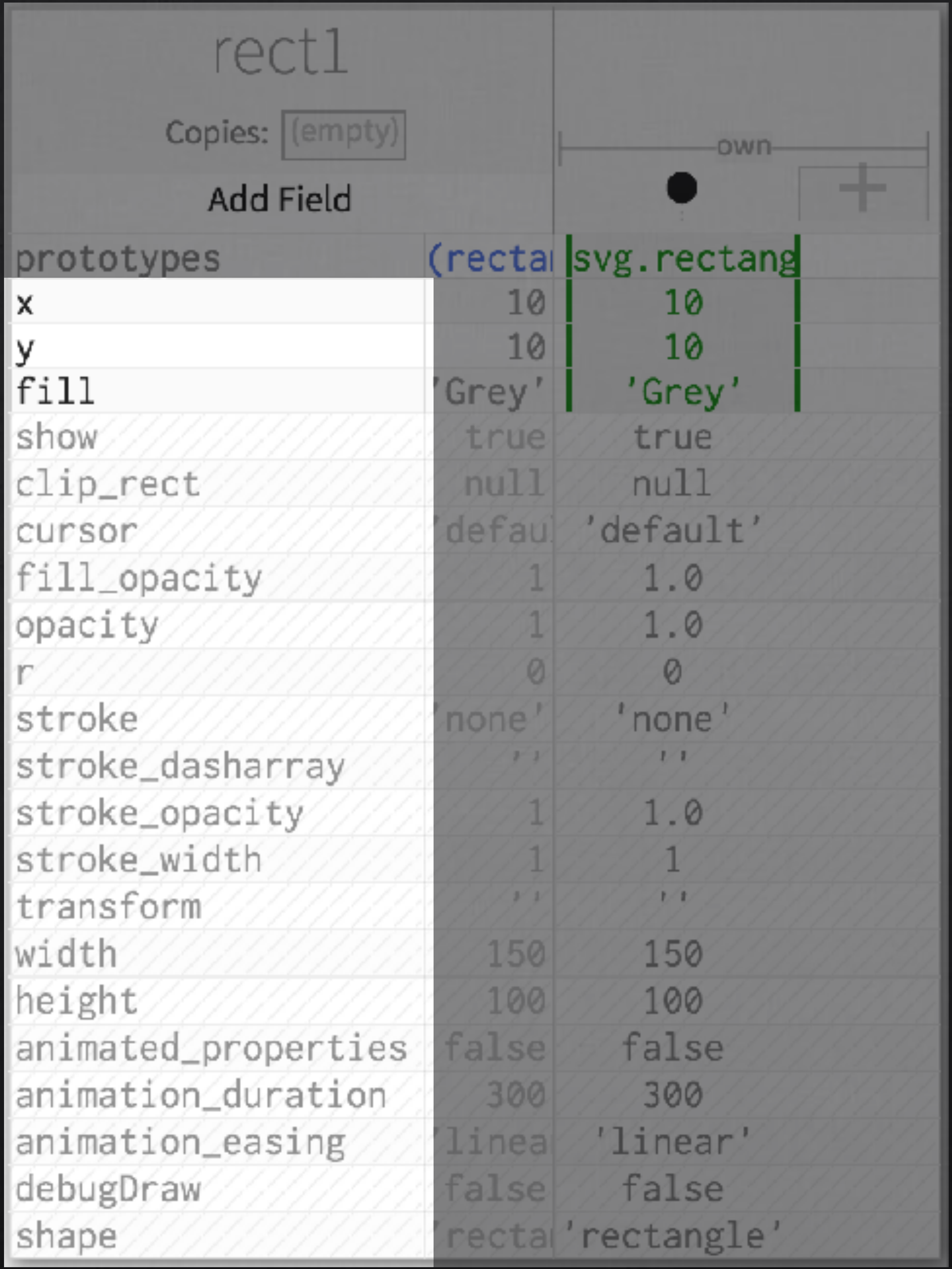


rect1

Copies: (empty)

Add Field

prototypes	(rectai	svg.rectang
x	10	10
y	10	10
fill	'Grey'	'Grey'
show	true	true
clip_rect	null	null
cursor	'defau	'default'
fill_opacity	1	1.0
opacity	1	1.0
r	0	0
stroke	'none'	'none'
stroke_dasharray	''	''
stroke_opacity	1	1.0
stroke_width	1	1
transform	''	''
width	150	150
height	100	100
animated_properties	false	false
animation_duration	300	300
animation_easing	'linea	'linear'
debugDraw	false	false
shape	'recta	'rectangle'

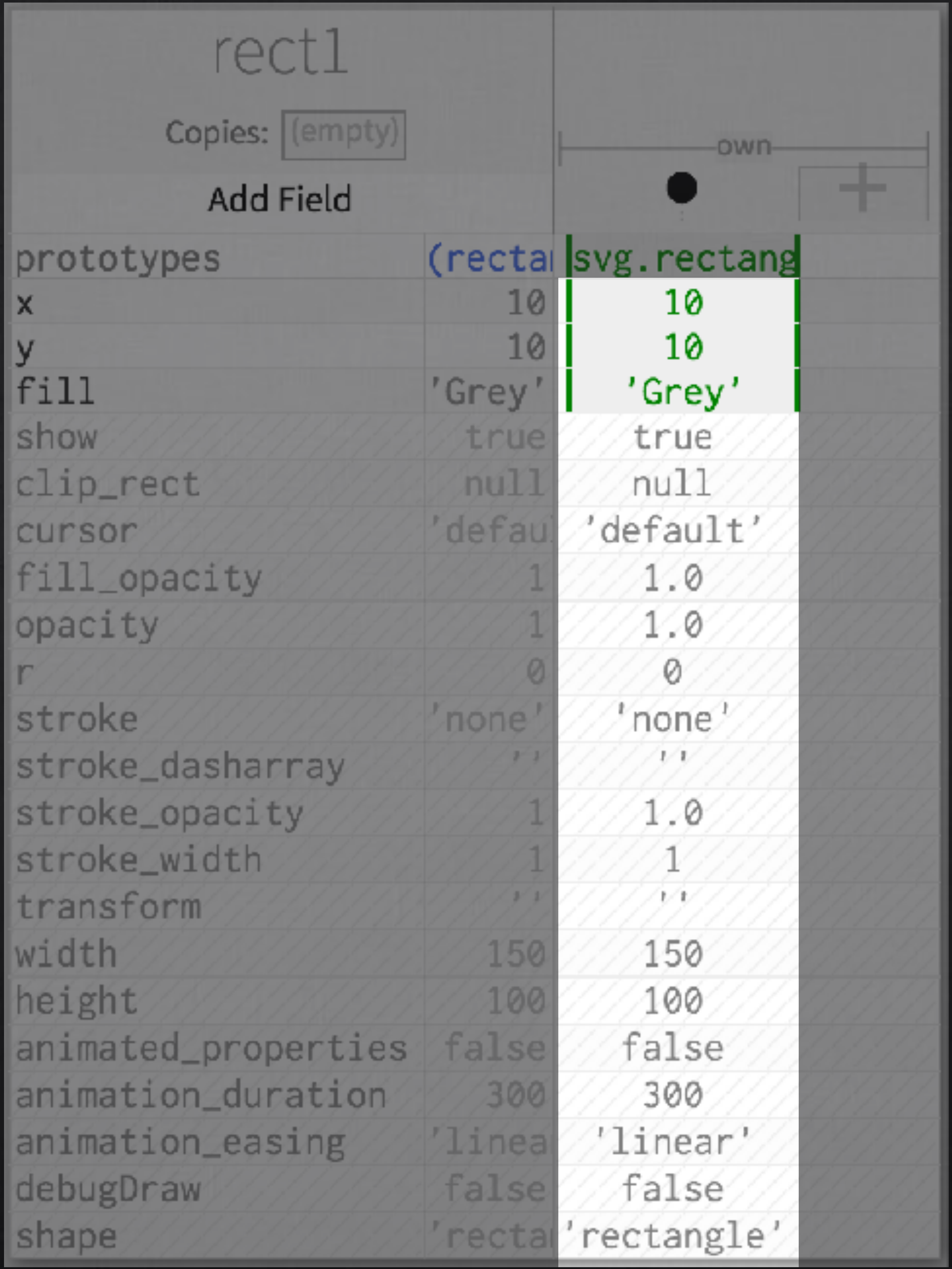


rect1

Copies: (empty)

Add Field

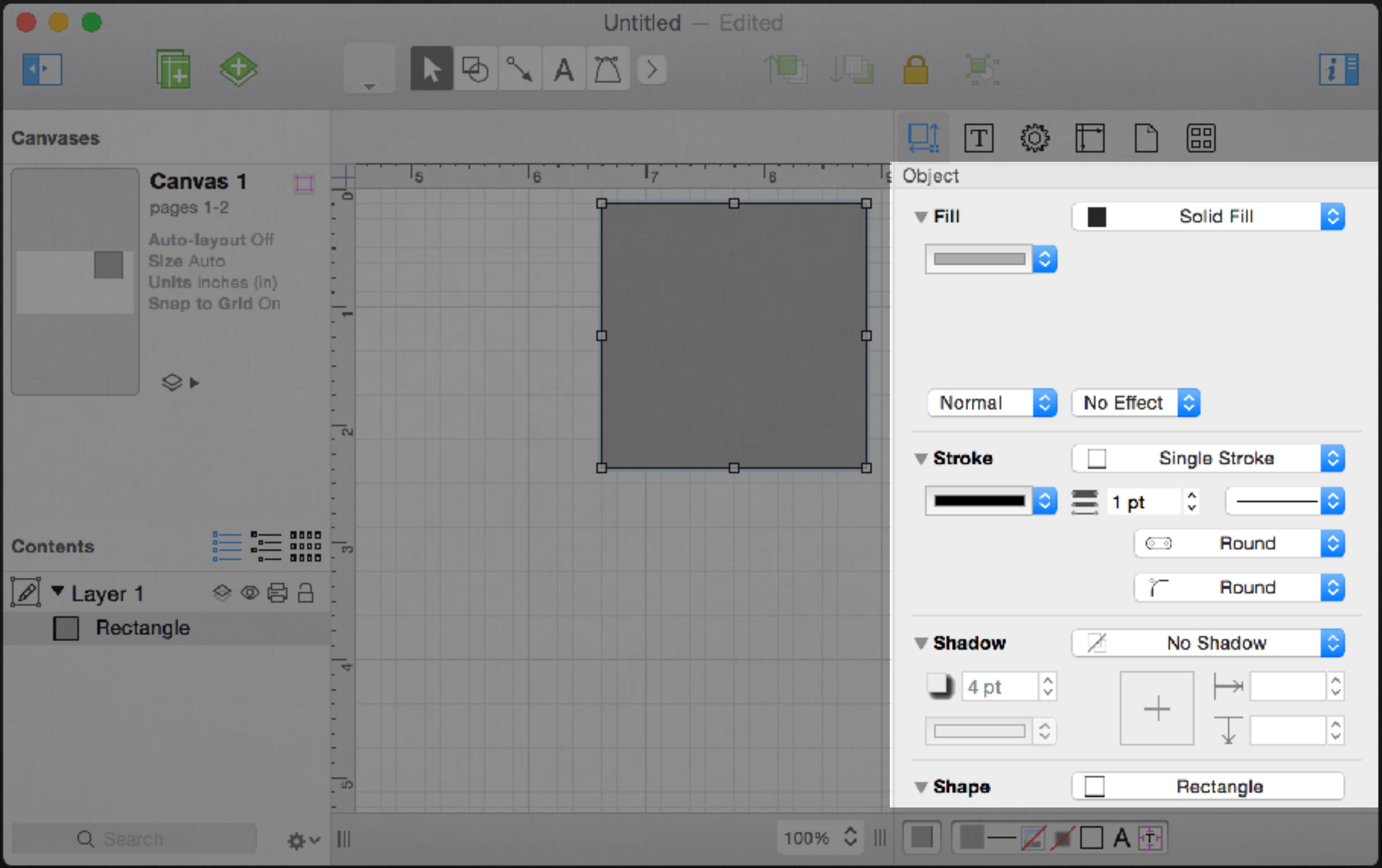
prototypes	(rectai	svg.rectang
x	10	10
y	10	10
fill	'Grey'	'Grey'
show	true	true
clip_rect	null	null
cursor	'defau	'default'
fill_opacity	1	1.0
opacity	1	1.0
r	0	0
stroke	'none'	'none'
stroke_dasharray	''	''
stroke_opacity	1	1.0
stroke_width	1	1
transform	''	''
width	150	150
height	100	100
animated_properties	false	false
animation_duration	300	300
animation_easing	'linea	'linear'
debugDraw	false	false
shape	'rectai	'rectangle'



rect1

Copies: (empty)

prototypes	(recta	svg.rectang
x	10	10
y	10	10
fill	'Grey'	'Grey'
show	true	true
clip_rect	null	null
cursor	'defau	'default'
fill_opacity	1	1.0
opacity	1	1.0
r	0	0
stroke	'none'	'none'
stroke_dasharray	''	''
stroke_opacity	1	1.0
stroke_width	1	1
transform	''	''
width	150	150
height	100	100
animated_properties	false	false
animation_duration	300	300
animation_easing	'linea	'linear'
debugDraw	false	false
shape	'recta	'rectangle'



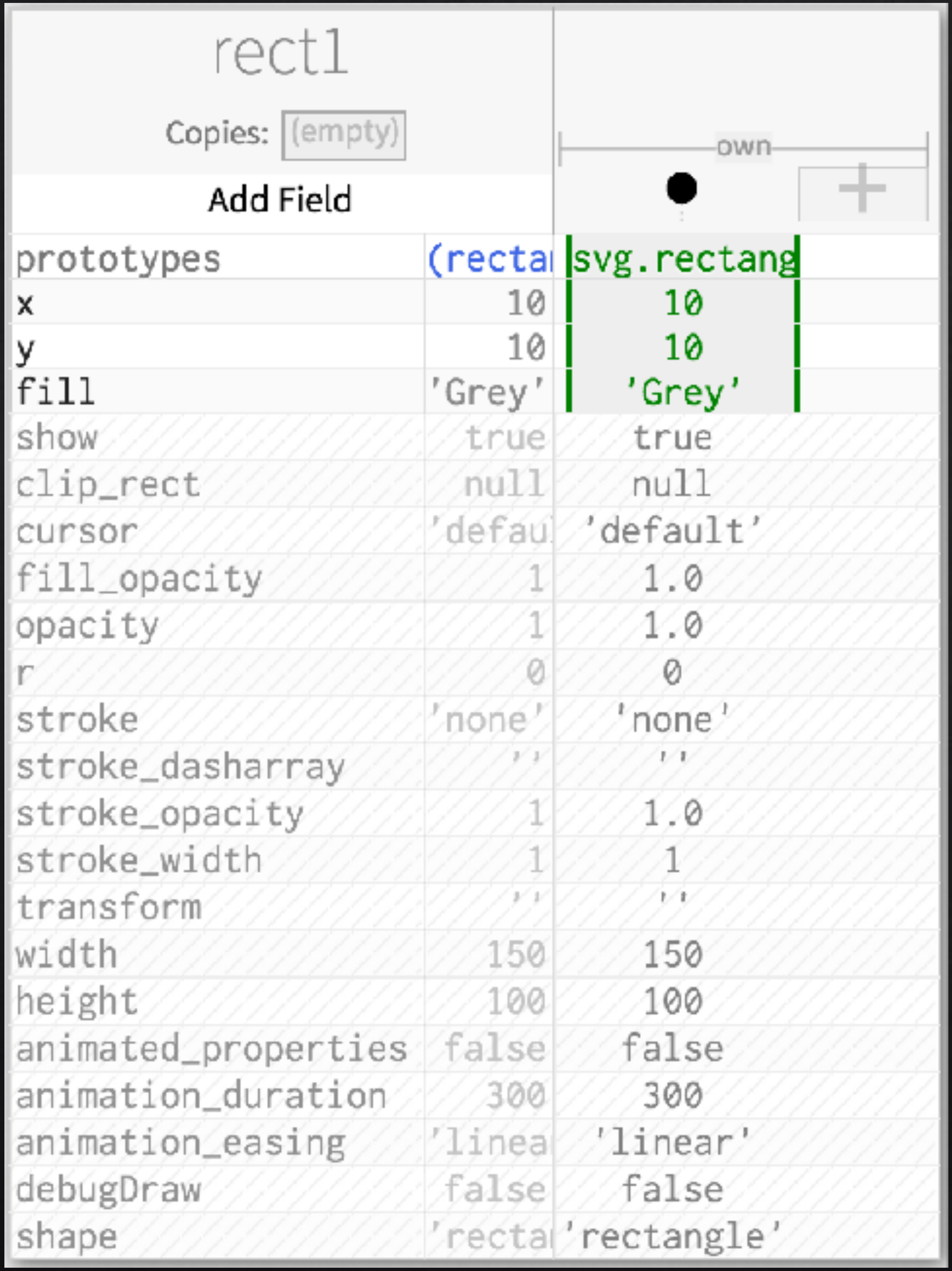
Canvas 1
pages 1-2
Auto-layout Off
Size Auto
Units inches (in)
Snap to Grid On

Fill: Solid Fill
Normal
No Effect

Stroke: Single Stroke
1 pt
Round
Round

Shadow: No Shadow
4 pt

Shape: Rectangle

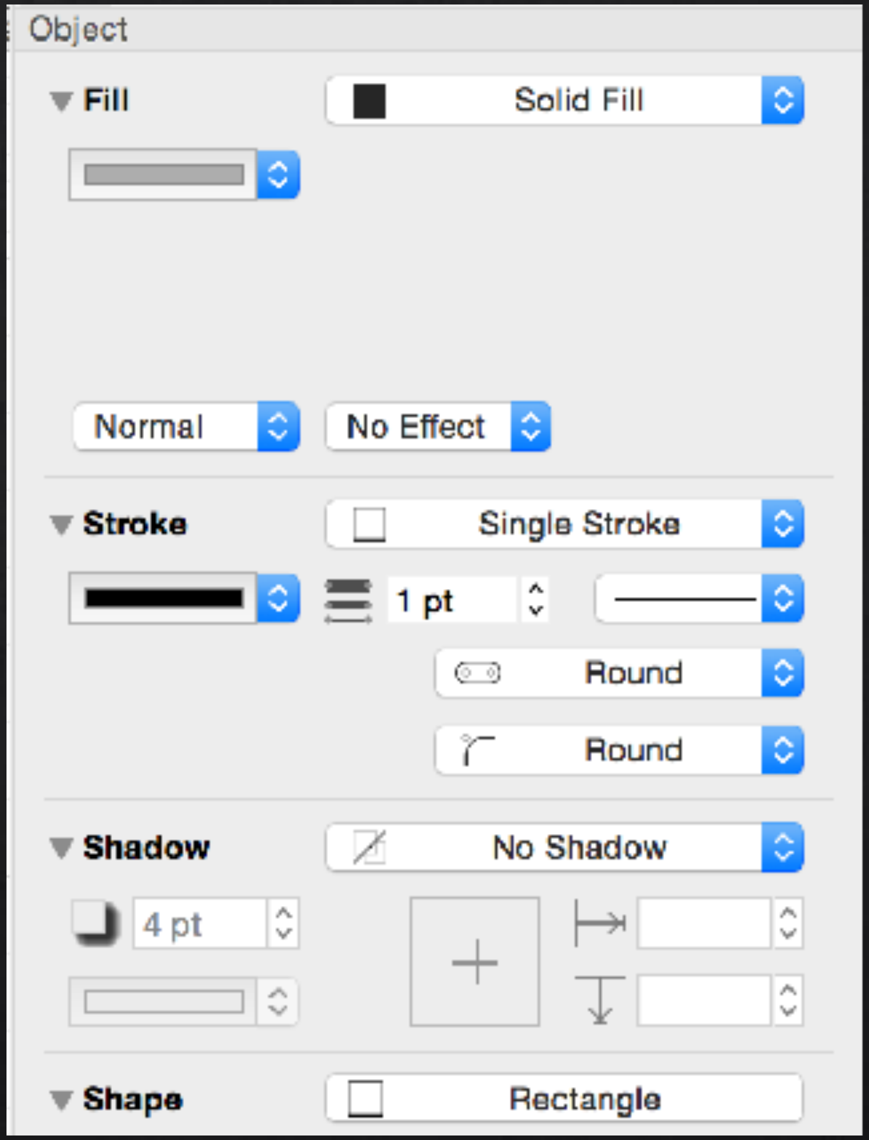


rect1

Copies: (empty)

Add Field

prototypes	(recta	svg.rectang
x	10	10
y	10	10
fill	'Grey'	'Grey'
show	true	true
clip_rect	null	null
cursor	'defau	'default'
fill_opacity	1	1.0
opacity	1	1.0
r	0	0
stroke	'none'	'none'
stroke_dasharray	''	''
stroke_opacity	1	1.0
stroke_width	1	1
transform	''	''
width	150	150
height	100	100
animated_properties	false	false
animation_duration	300	300
animation_easing	'linea	'linear'
debugDraw	false	false
shape	'recta	'rectangle'



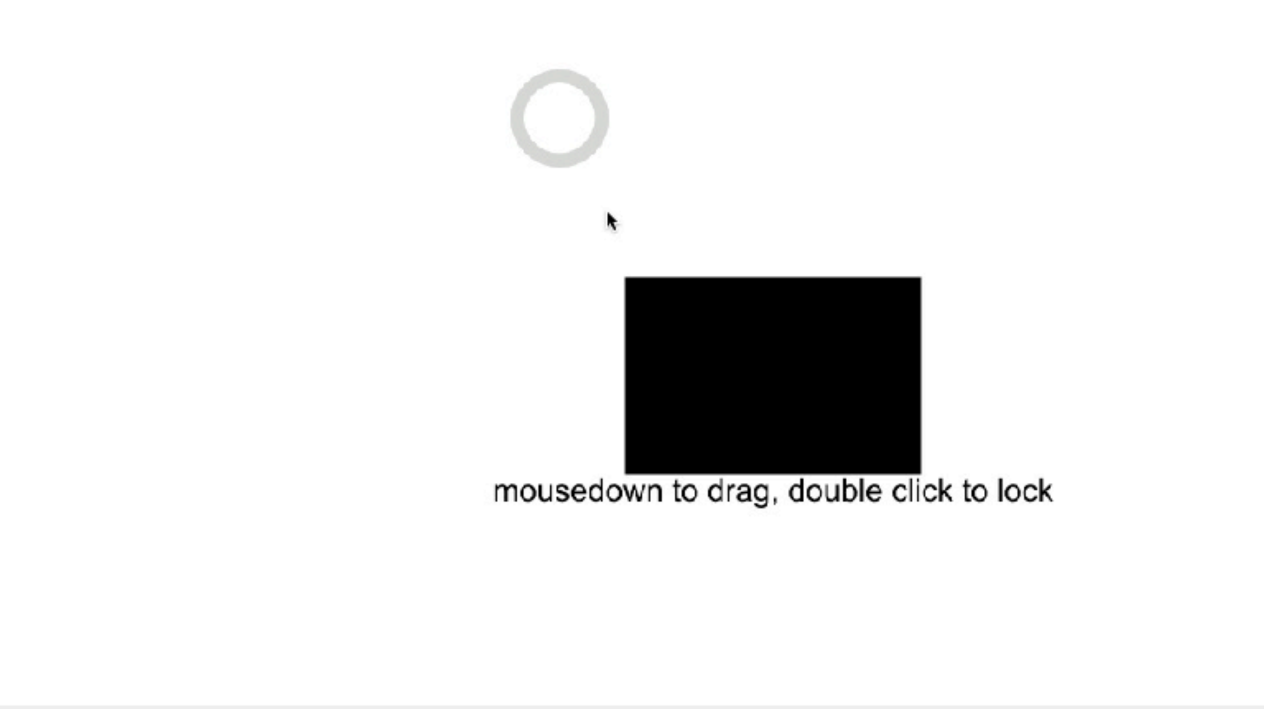
Object

- Fill: Solid Fill
- Normal
- No Effect
- Stroke: Single Stroke
- 1 pt
- Round
- Round
- Shadow: No Shadow
- 4 pt
- Shape: Rectangle

user evaluation

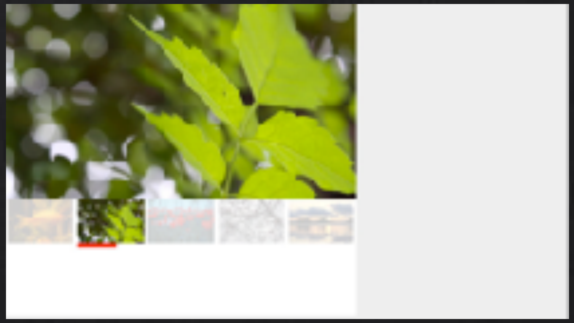
- **hypothesis**: easier to understand/modify InterState code than event-callback code
- 20 participants
  - (some programming experience)

drag lock

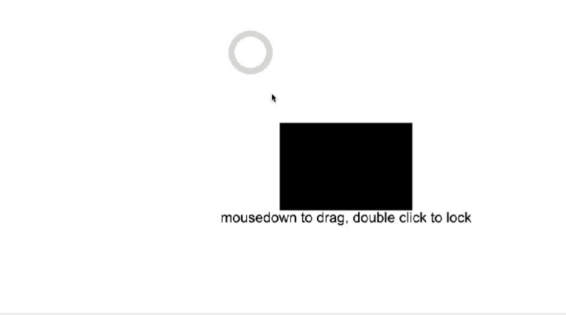


mousedown to drag, double click to lock

image carousel

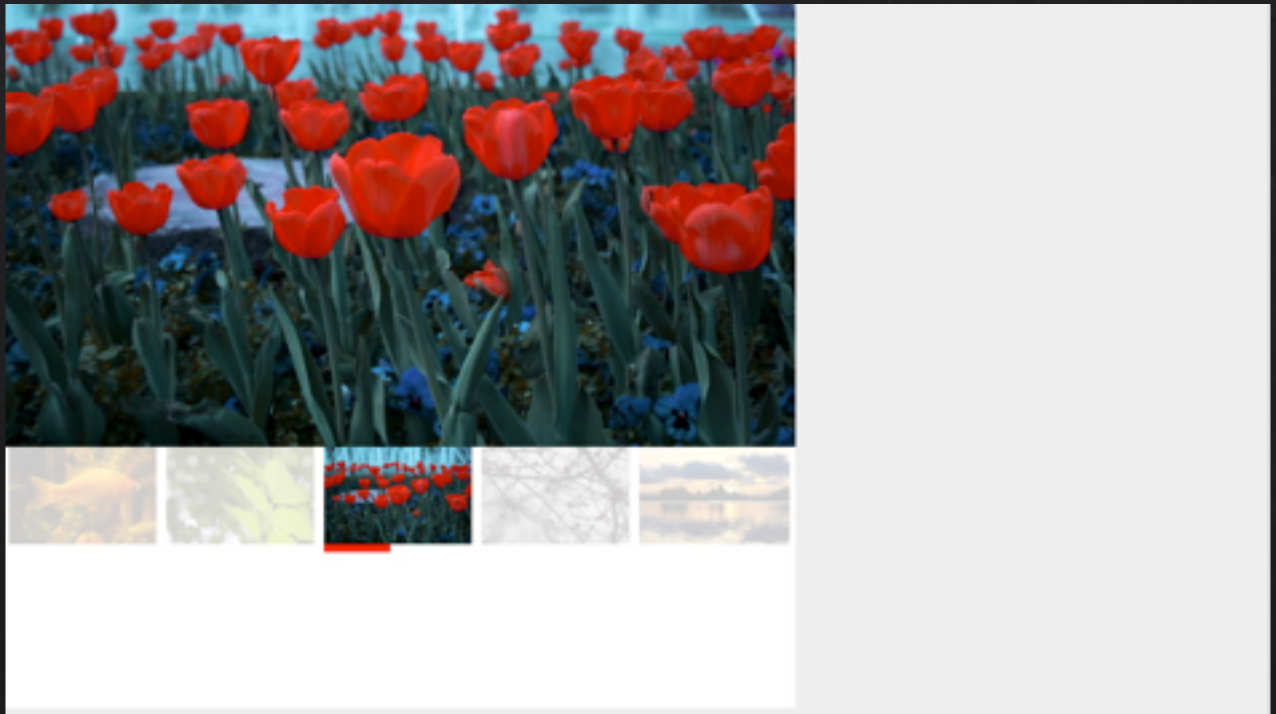


drag lock



mousedown to drag, double click to lock

image carousel



drag lock

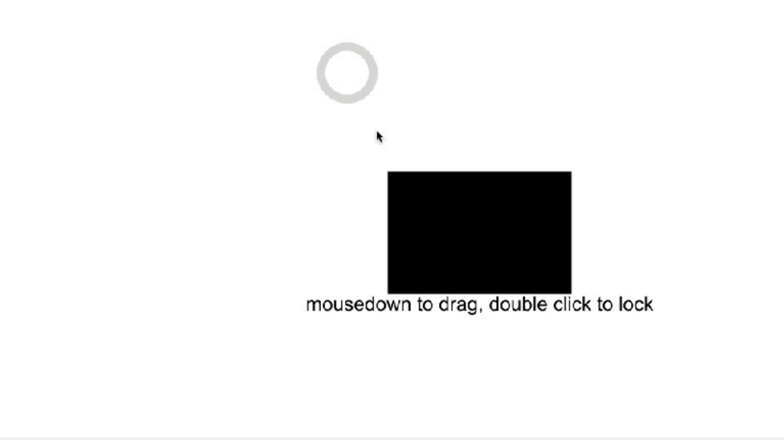
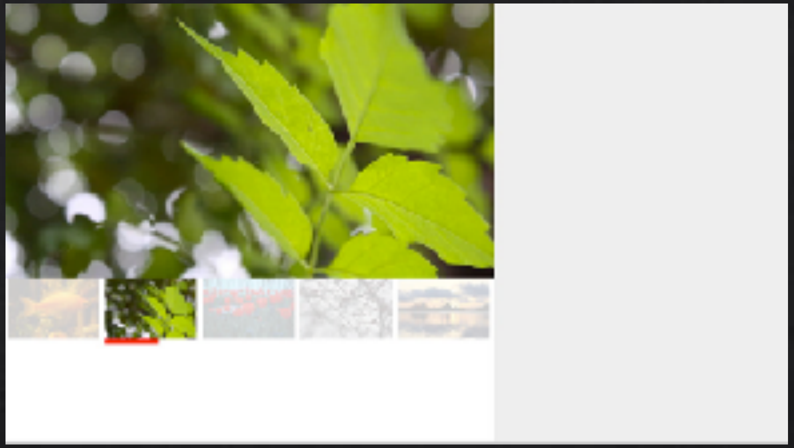
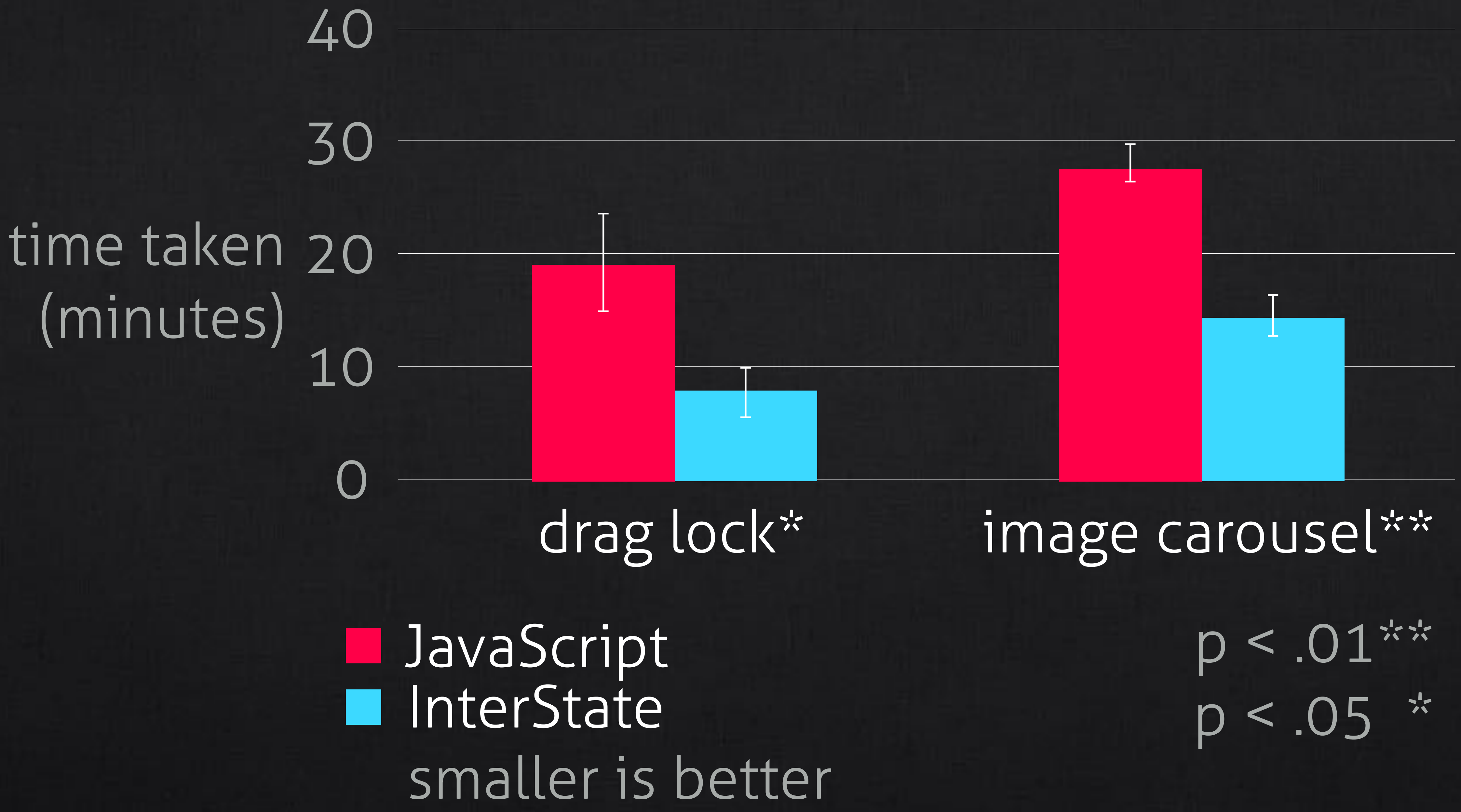

mousedown to drag, double click to lock

image carousel

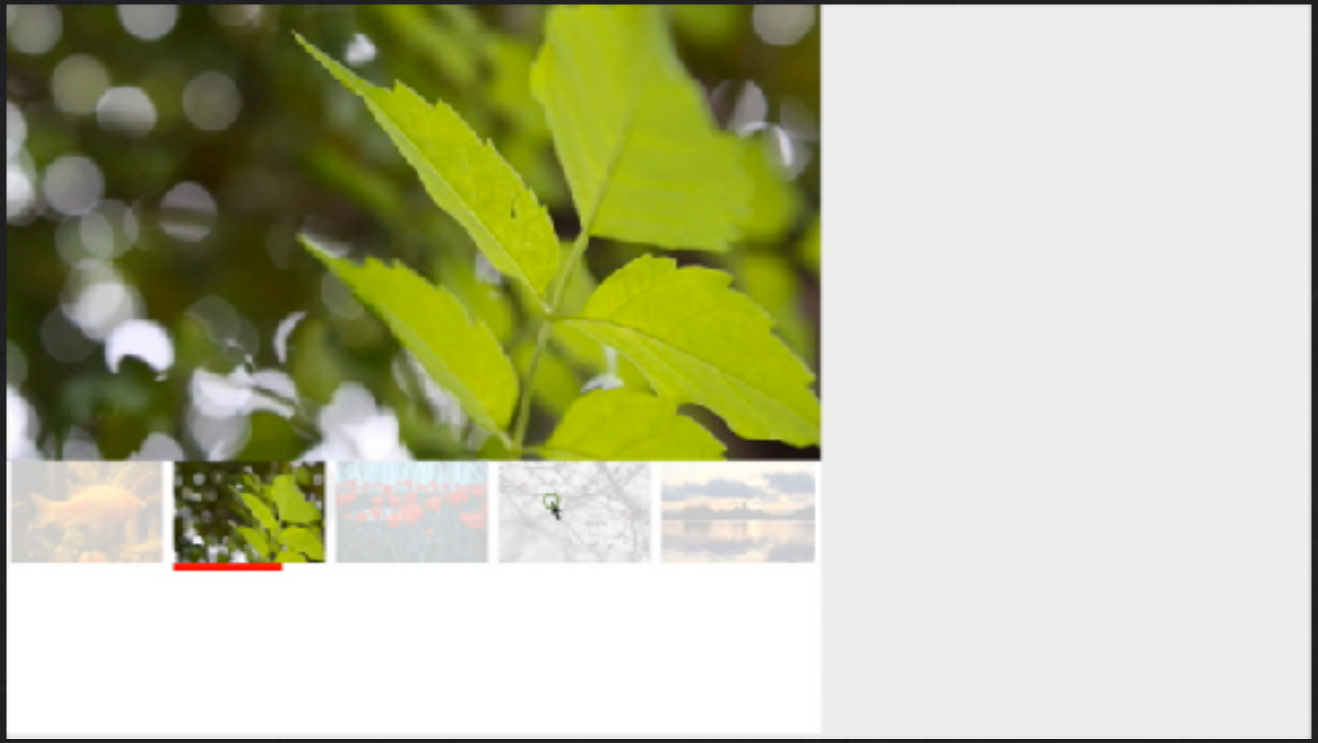


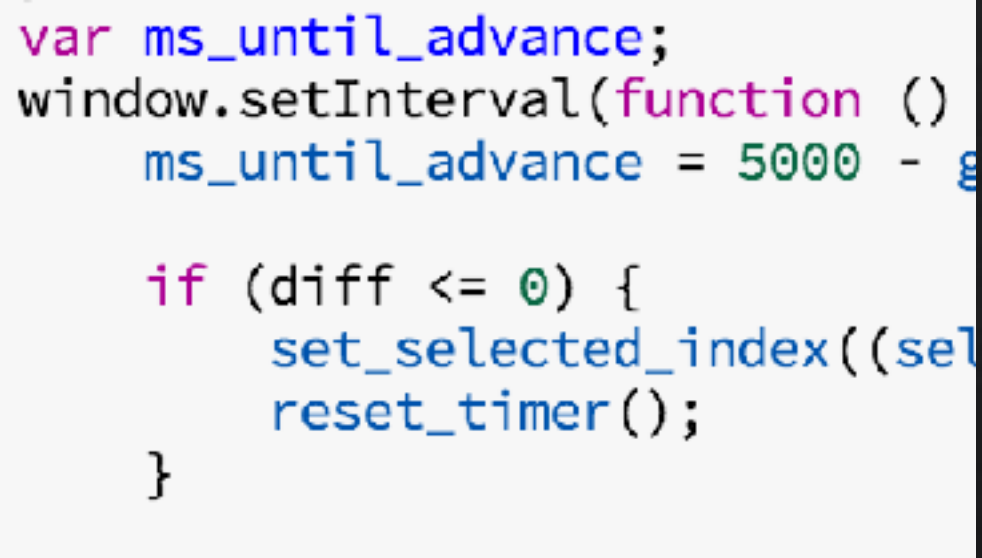


smaller is better

p < .01**

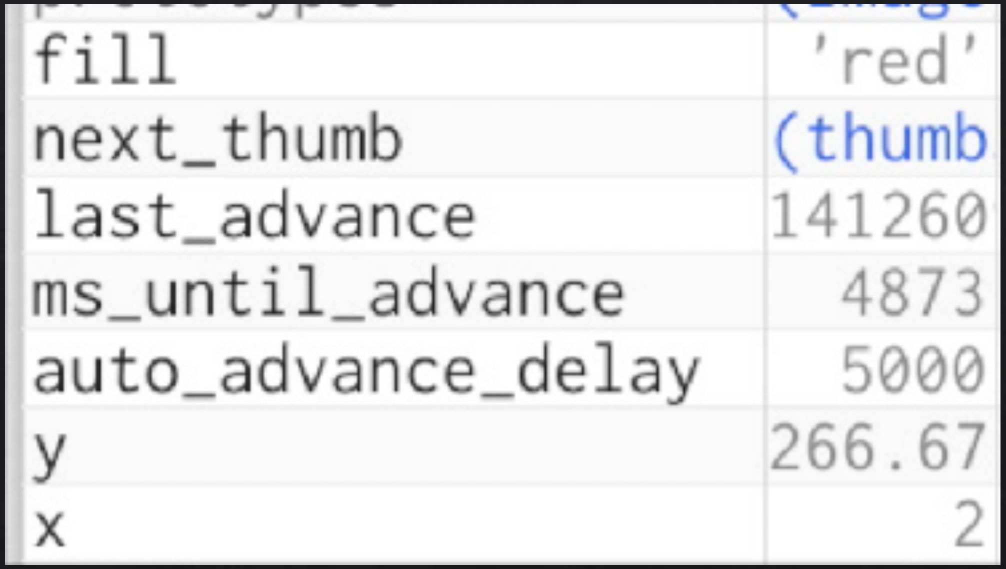
p < .05 *





```
var ms_until_advance;
window.setInterval(function ()
    ms_until_advance = 5000 - g

    if (diff <= 0) {
        set_selected_index((sel
        reset_timer();
    }
```

JavaScript



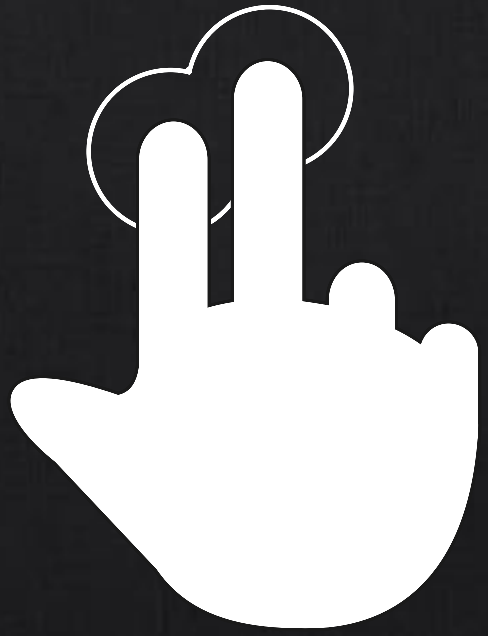
fill	'red'
next_thumb	(thumb
last_advance	141260
ms_until_advance	4873
auto_advance_delay	5000
y	266.67
x	2

InterState

multi-touch

- larger state space for gestures
  - delay between events and recognition
- richer gesture features
- expectations of directness

touch clusters



num_fingers = 2

touch clusters

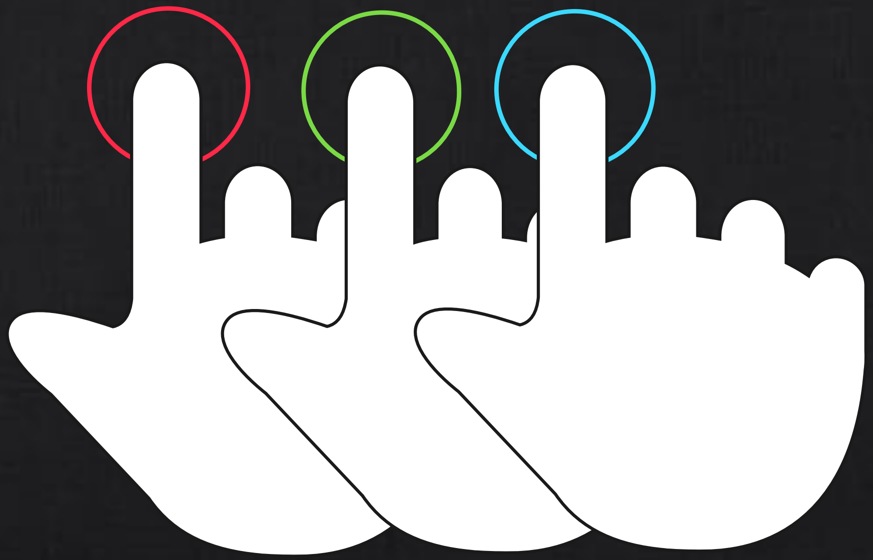
- descriptive touch events
- developers specify number of fingers, where pressed, etc
- outputs: location, scale, rotation

resolving conflicts

- optional delay & priority for events
- touch clusters can determine whether other clusters can use same touches

resolving conflicts

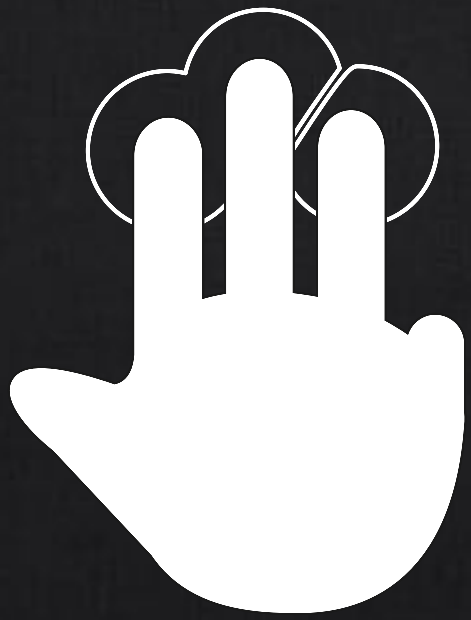
three-finger
cluster fires

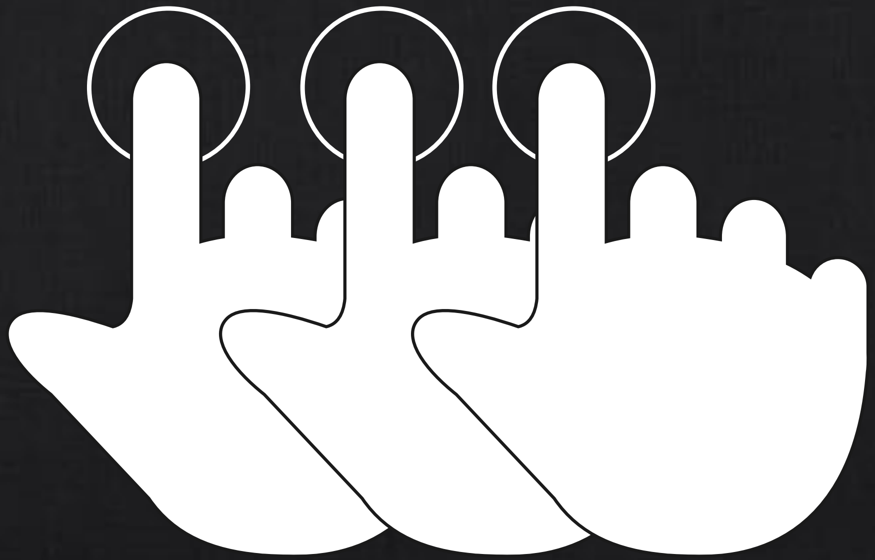


three single-touch
clusters fire

resolving conflicts

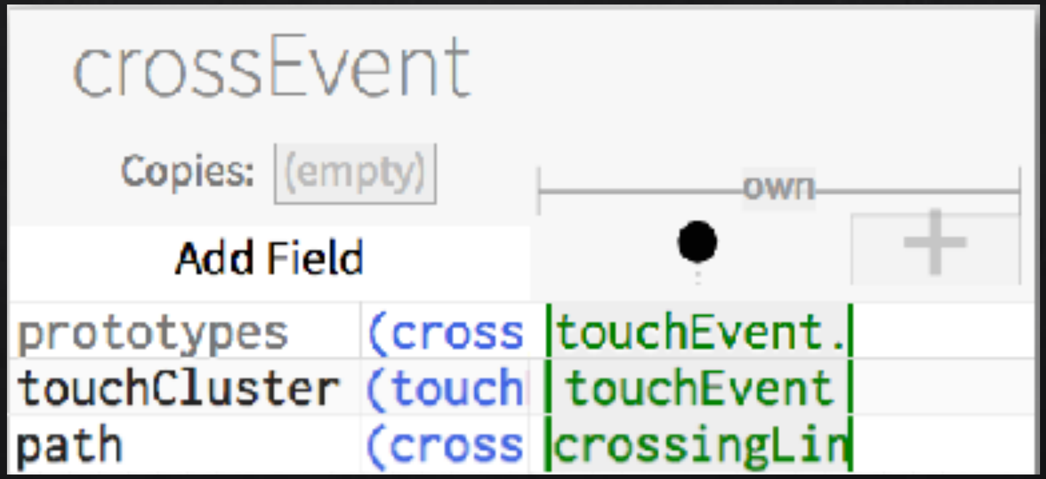
resolving conflicts





delay: 100ms

crossing events

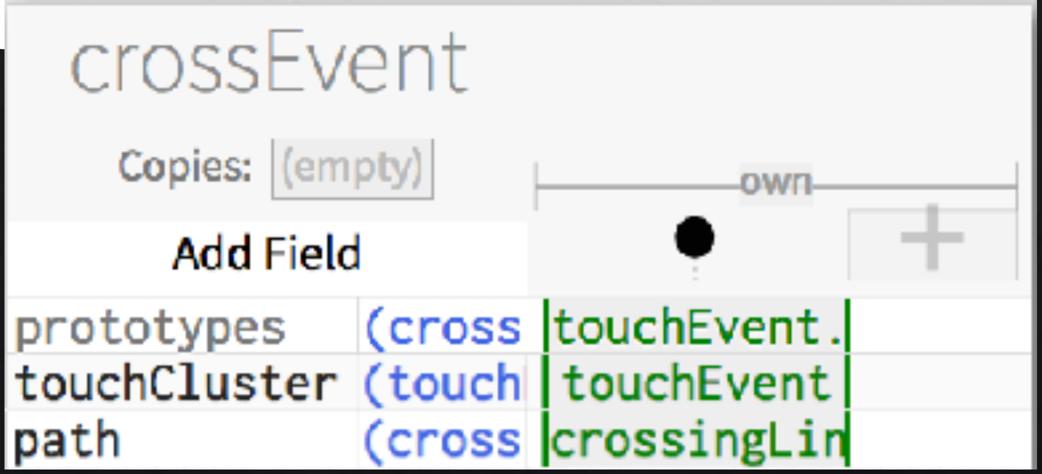


crossEvent

Copies: (empty)

prototypes	(cross	touchEvent.
touchCluster	(touch	touchEvent
path	(cross	crossingLin

crossing events



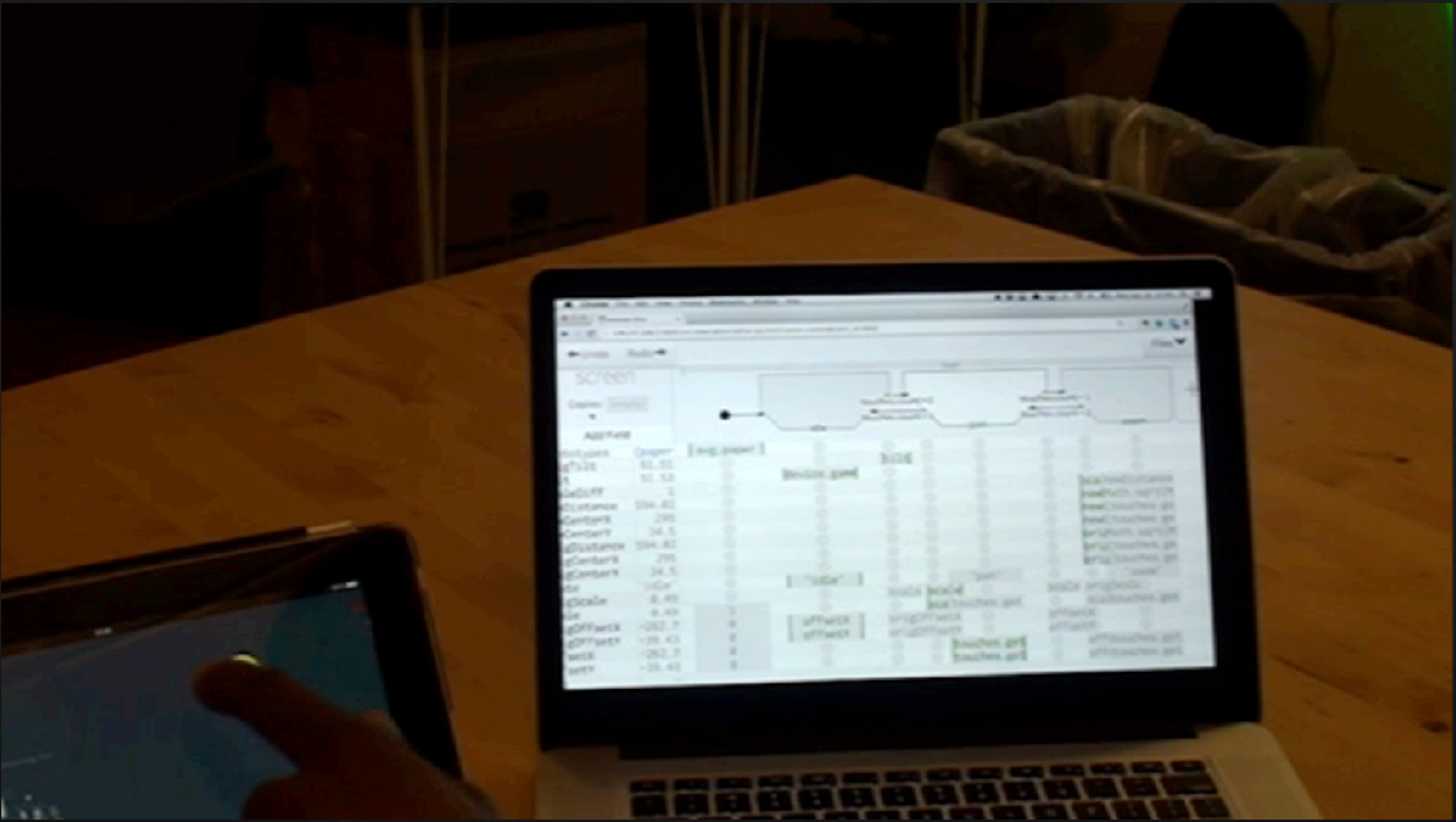
crossEvent

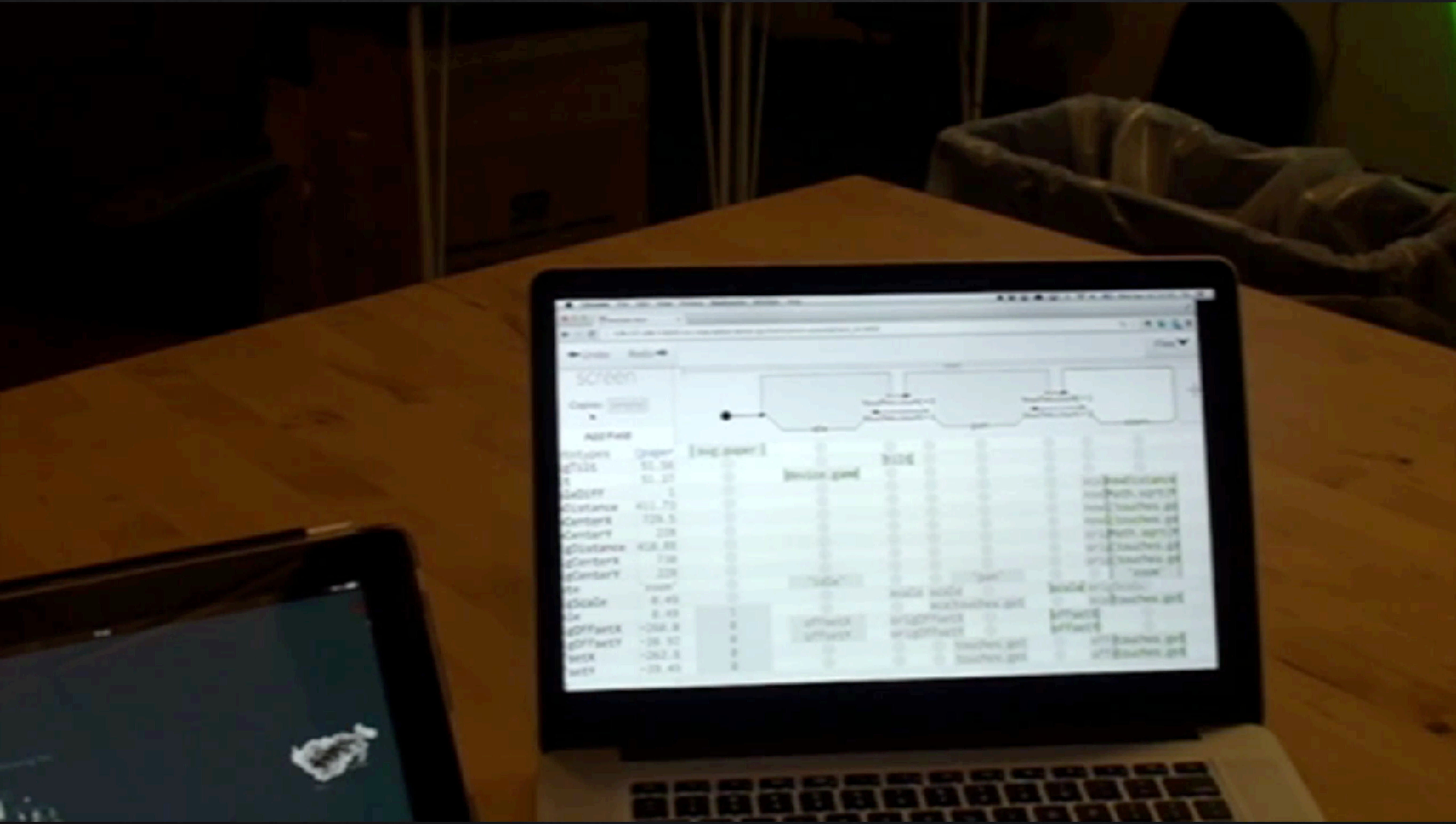
Copies: (empty)

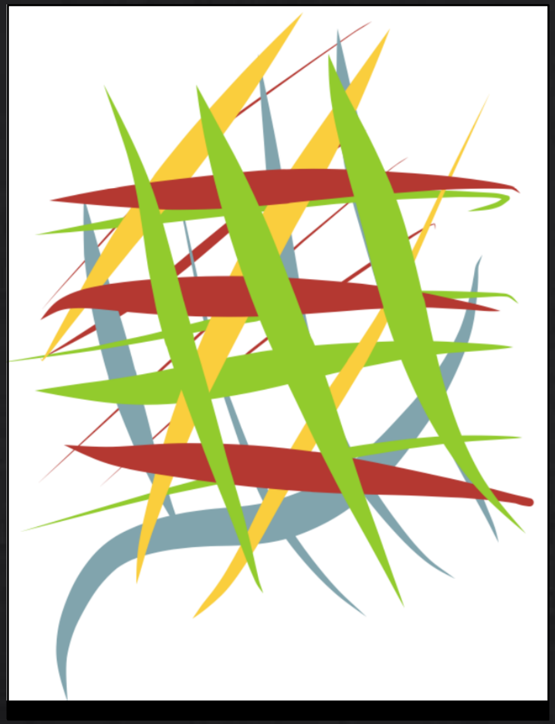
prototypes	(cross	touchEvent.
touchCluster	(touch	touchEvent
path	(cross	crossingLin

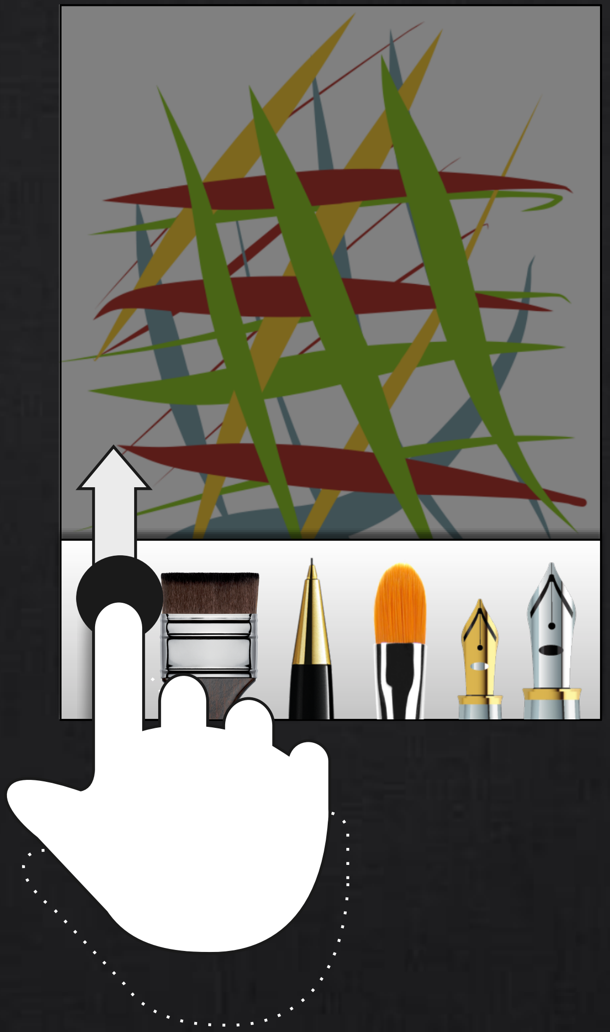
InterState + touch

- descriptive touch events
- natural constraints on finger properties
- manage conflicts between touchscreen gestures
- mechanisms for event abstraction

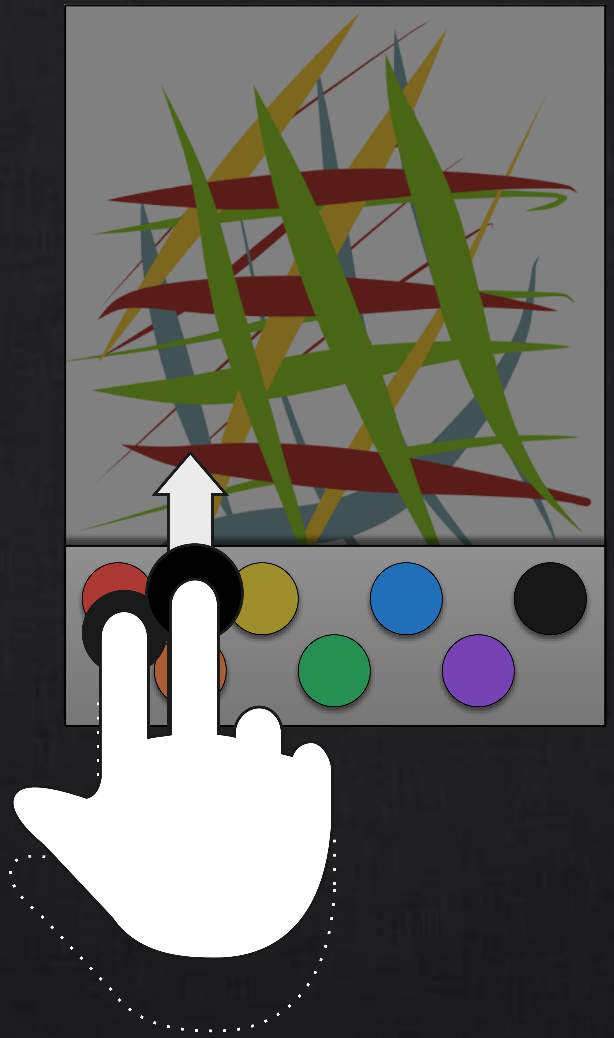

Undo
Redo
screen
Add Field


Undo
Redo
screen
Add Field



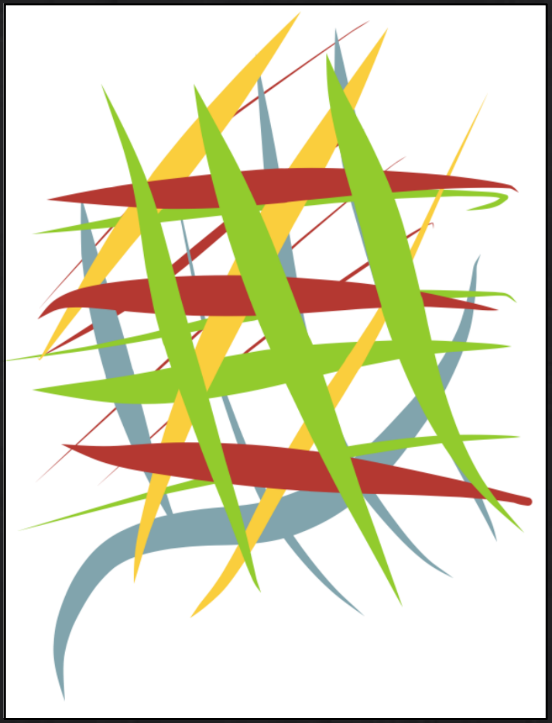


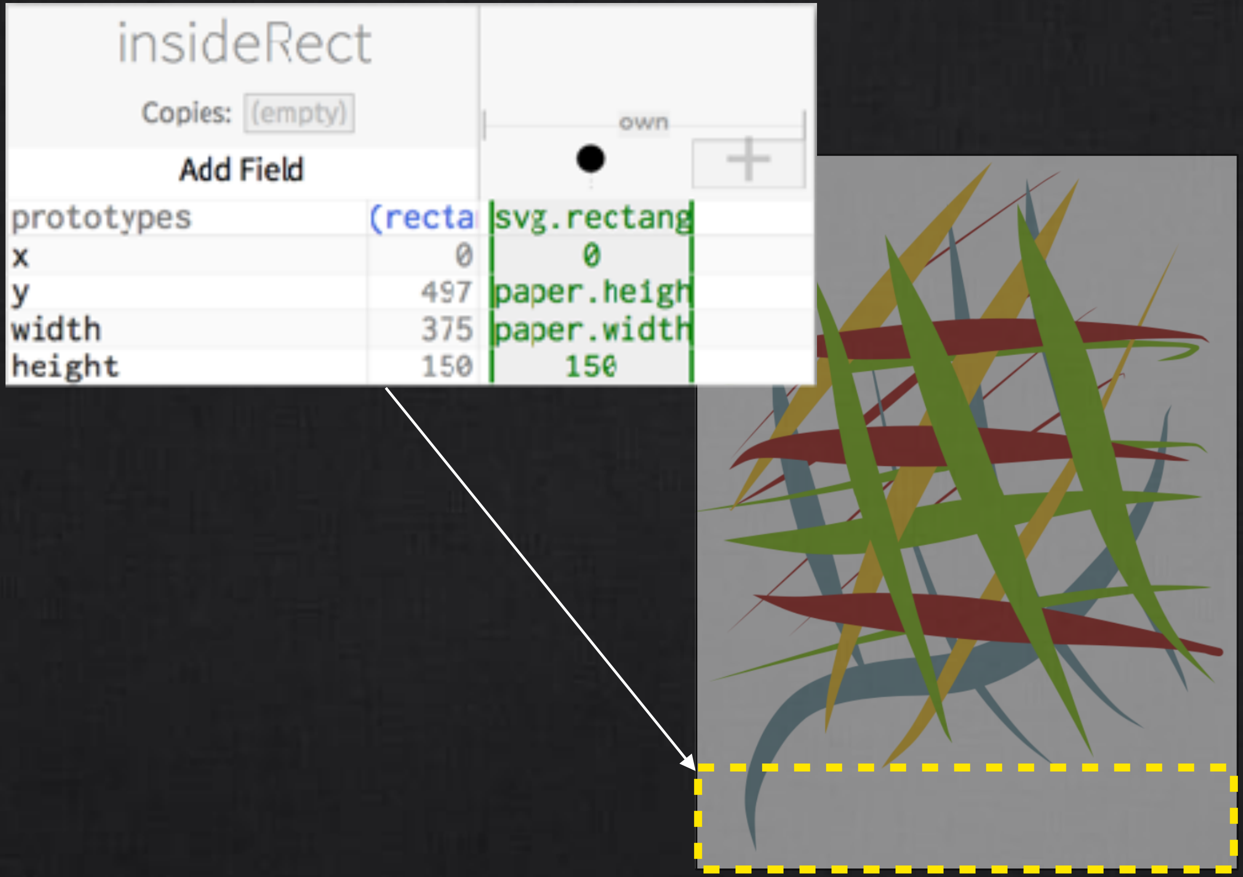
a one-finger swipe
from the bottom brings
up the brush menu



a two-finger swipe
from the bottom brings
up the color menu

Sprint
14:30
EDIT

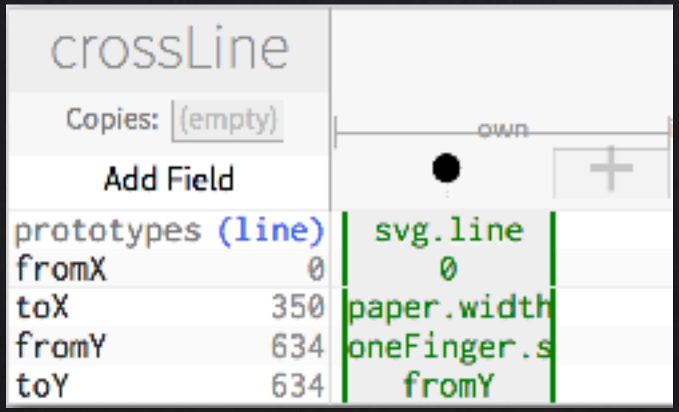




insideRect

Copies: (empty)

prototypes	(recta	svg.rectang
x	0	0
y	497	paper.heigh
width	375	paper.width
height	150	150

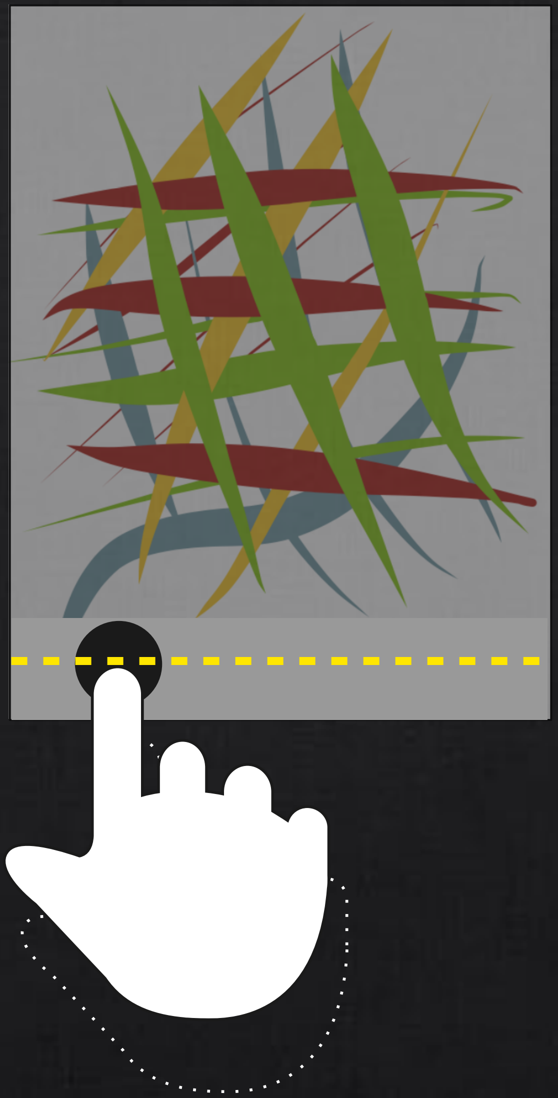


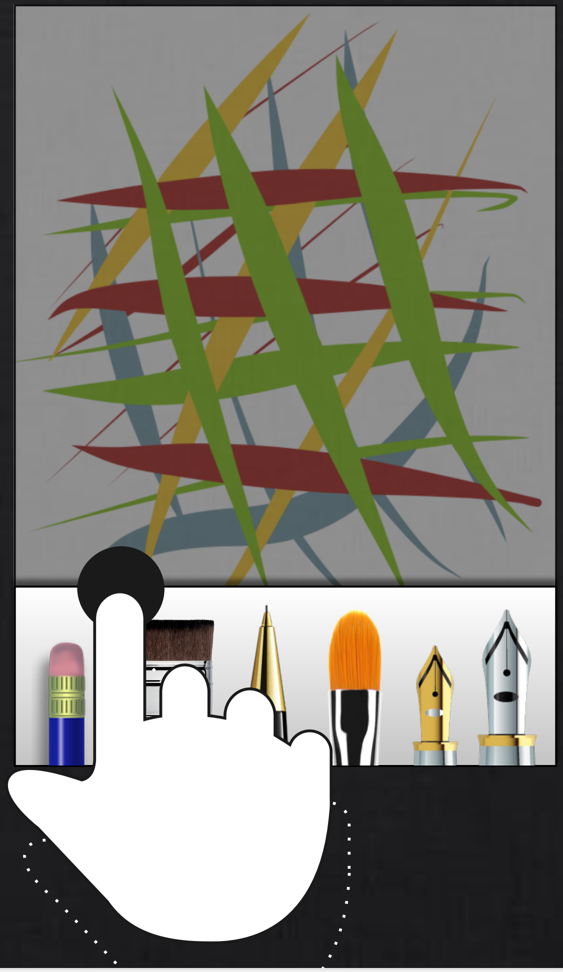
crossLine

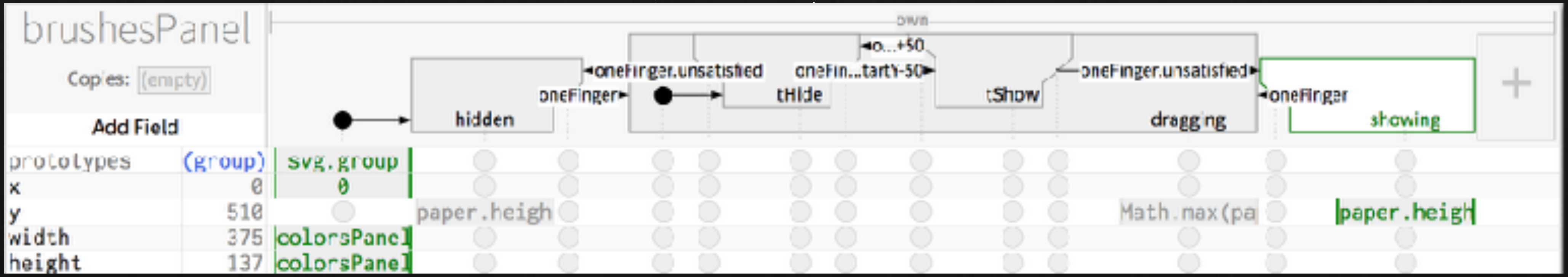
Copies: (empty)

Add Field

prototypes (line)		svg.line
fromX	0	0
toX	350	paper.width
fromY	634	oneFinger.s
toY	634	fromY





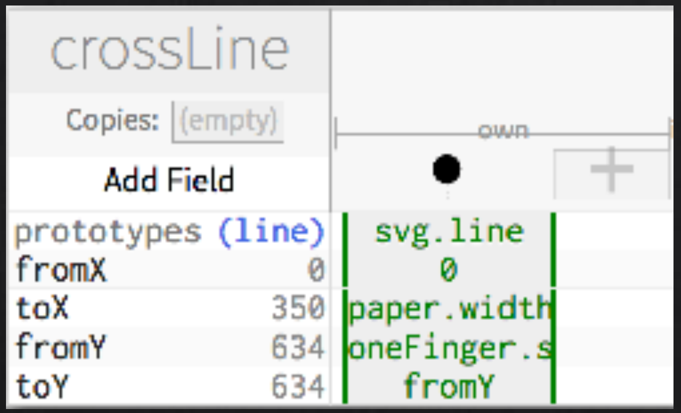


brushesPanel

Copies: (empty)

Add Field

prototypes	(group)	svg.group			
x	0	0			
y	510		paper.heigh	Math.max(pa	paper.heigh
width	375	colorsPanel			
height	137	colorsPanel			



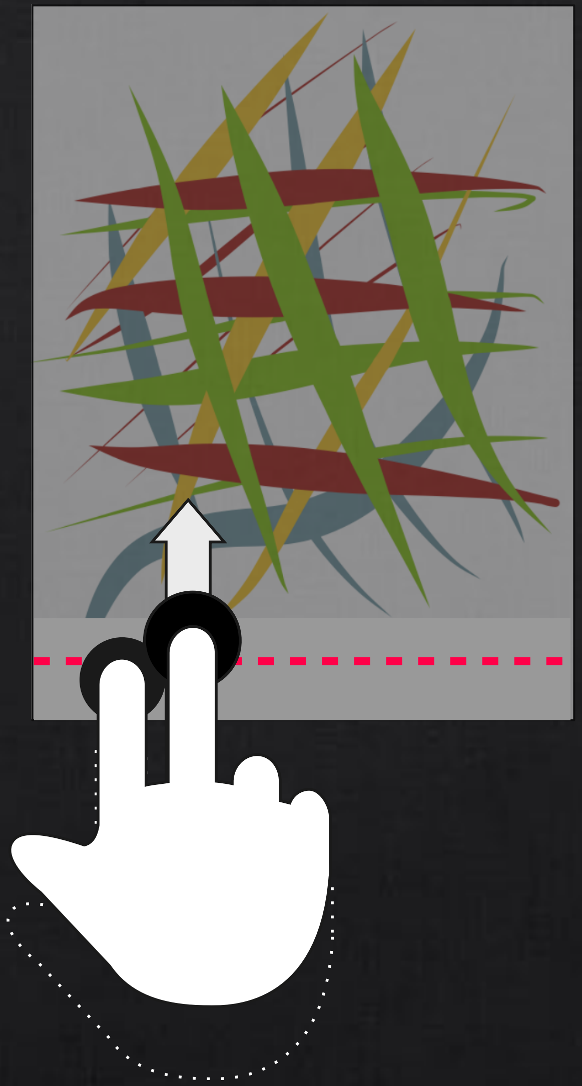
crossLine

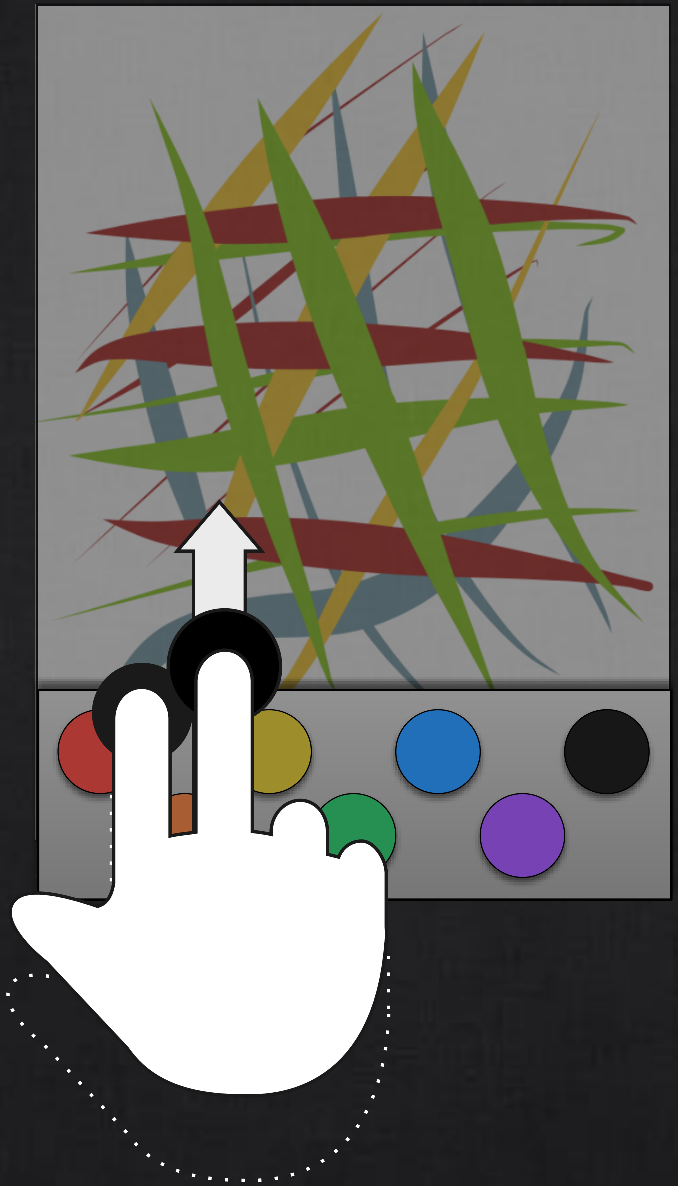
Copies: (empty)

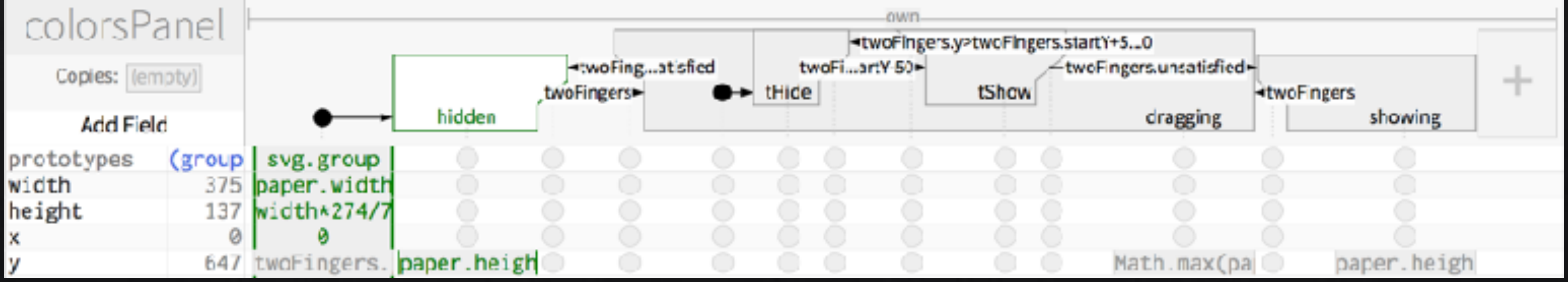
Add Field

● +

prototypes (line)		svg.line
fromX	0	0
toX	350	paper.width
fromY	634	oneFinger.s
toY	634	fromY





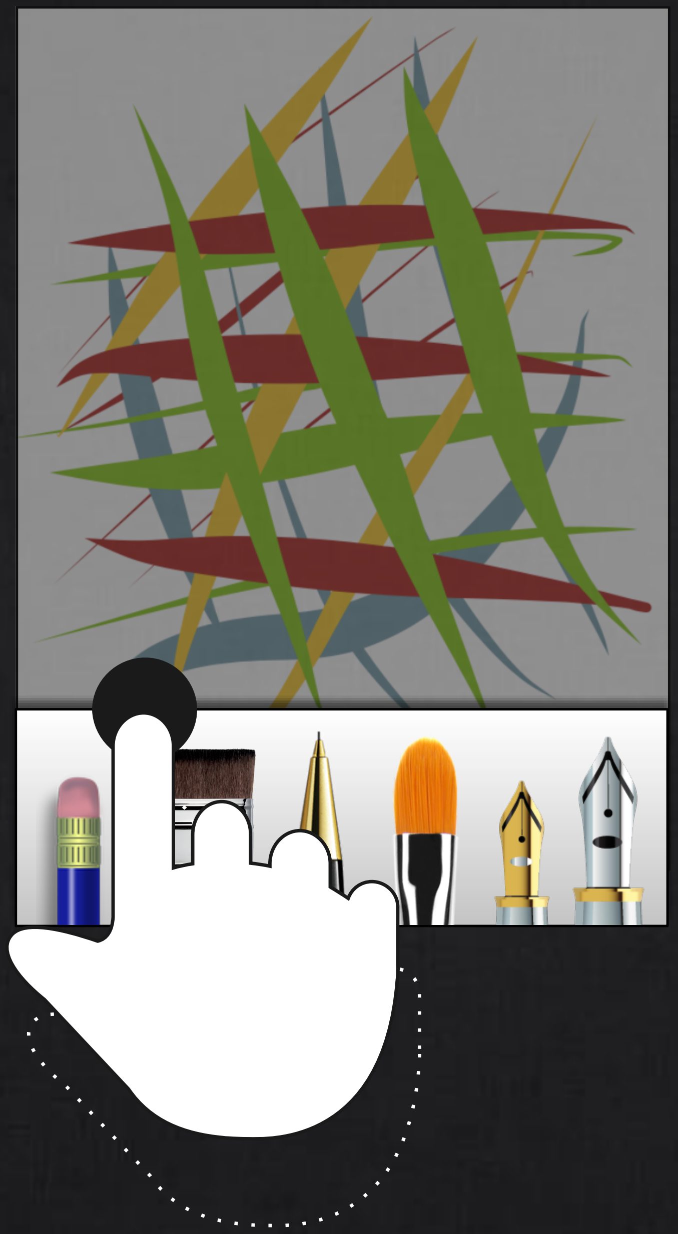


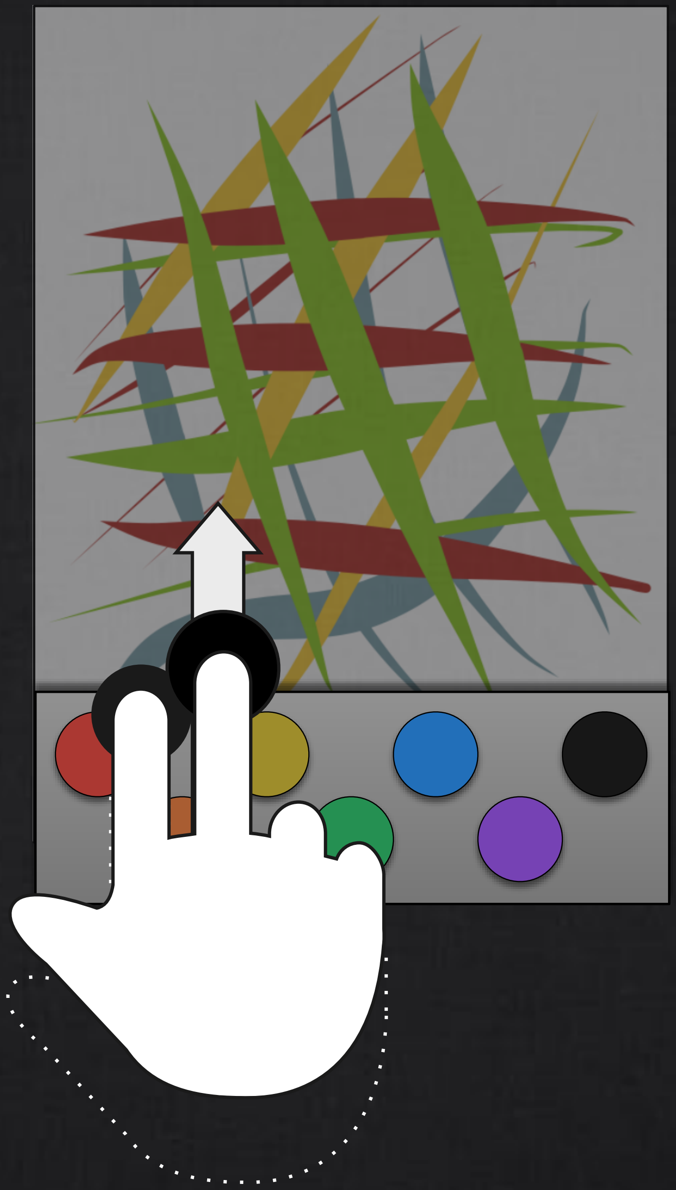
colorsPanel

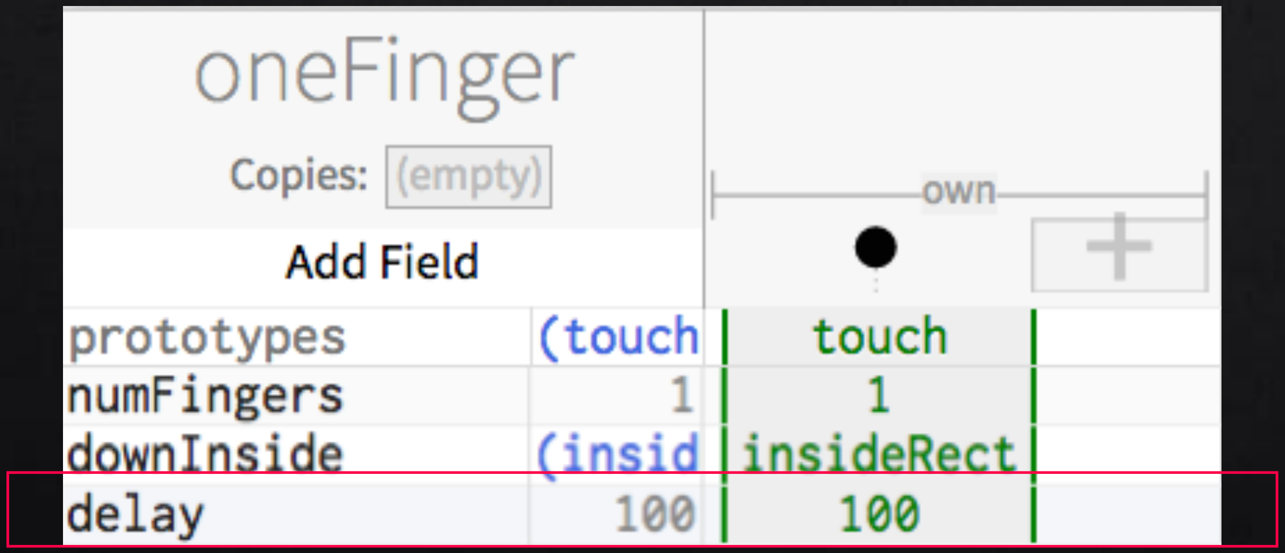
Copies: (empty)

Add Field

prototypes	(group	svg.group			
width	375	paper.width			
height	137	width*274/7			
x	0	0			
y	647	twoFingers.	paper.heigh	Math.max(pa	paper.heigh





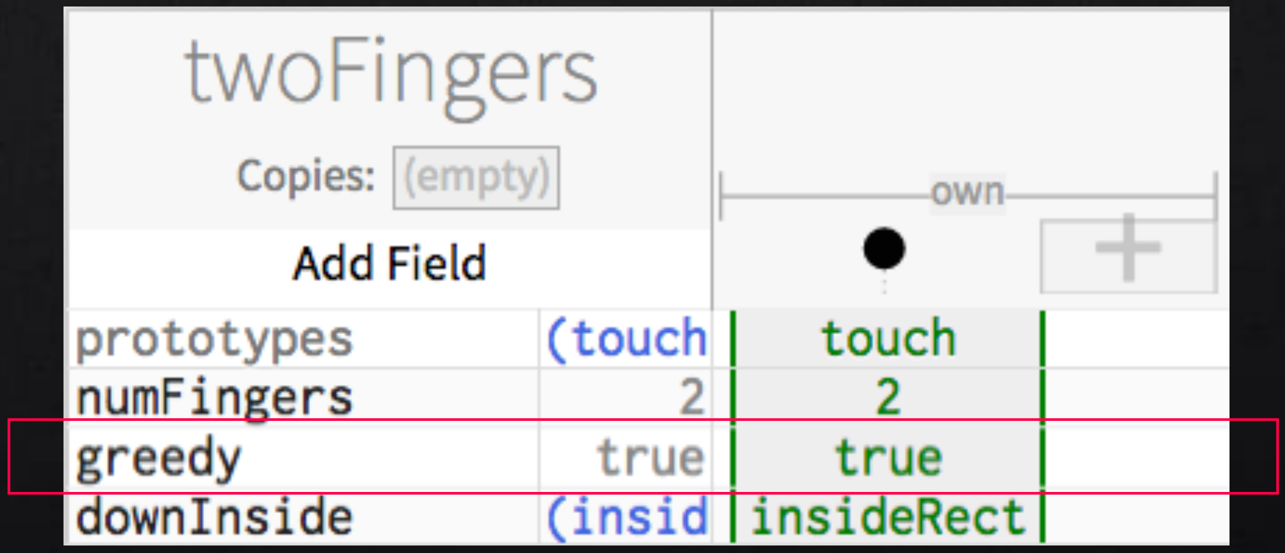


oneFinger

Copies: (empty)

Add Field

prototypes	(touch	touch
numFingers	1	1
downInside	(insid	insideRect
delay	100	100



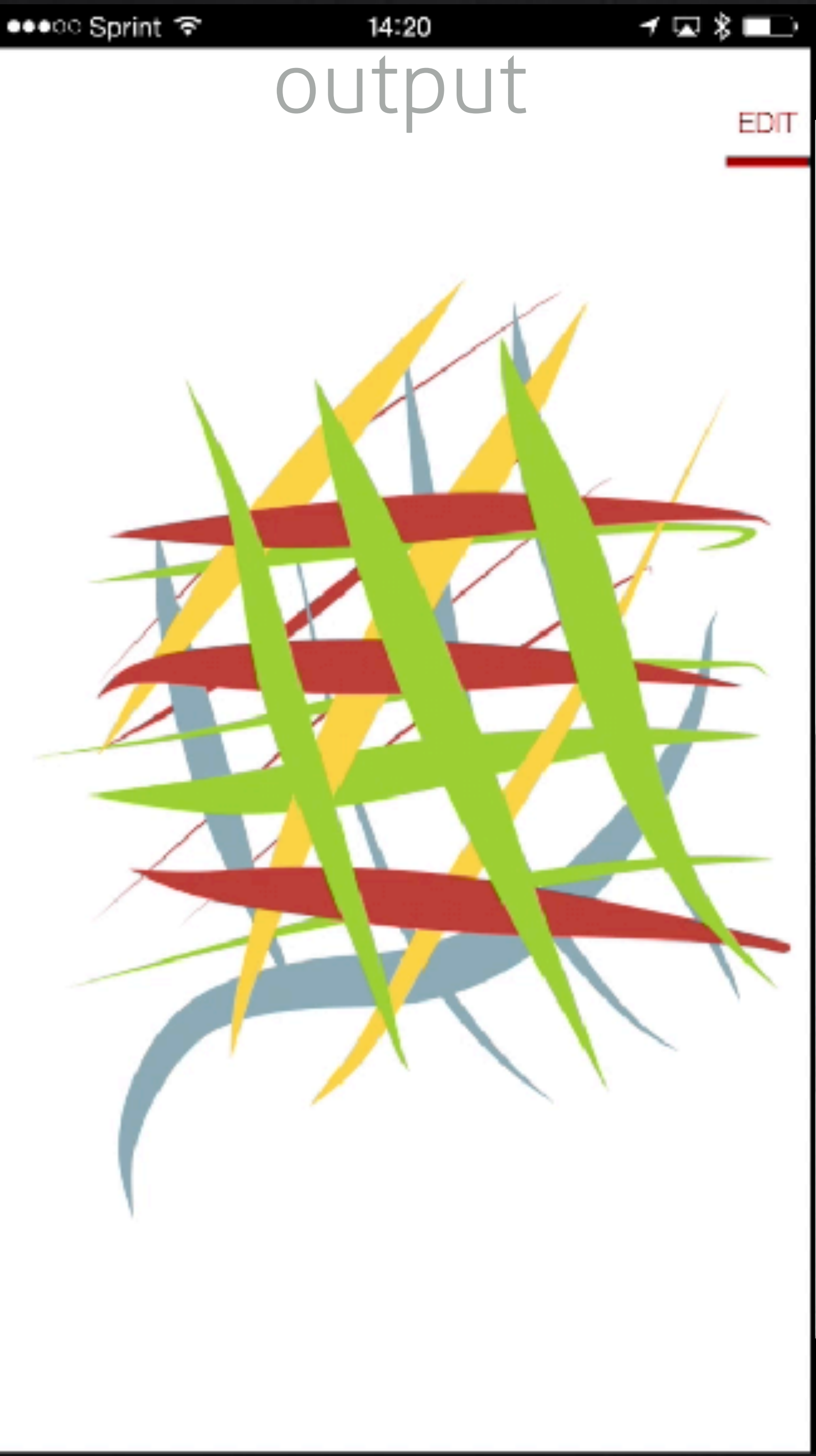
twoFingers

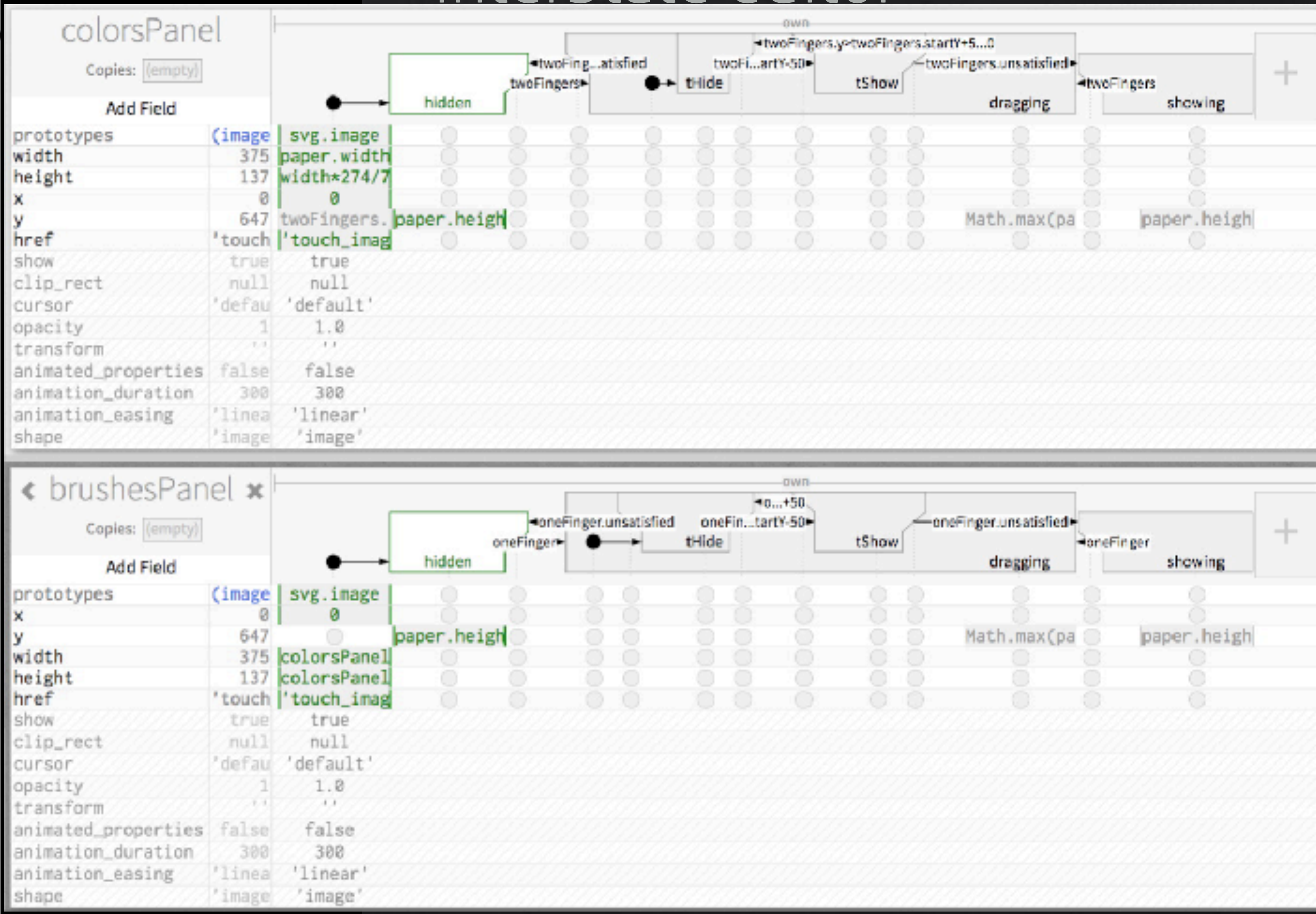
Copies: (empty)

Add Field

prototypes	(touch	touch
numFingers	2	2
greedy	true	true
downInside	(insid	insideRect

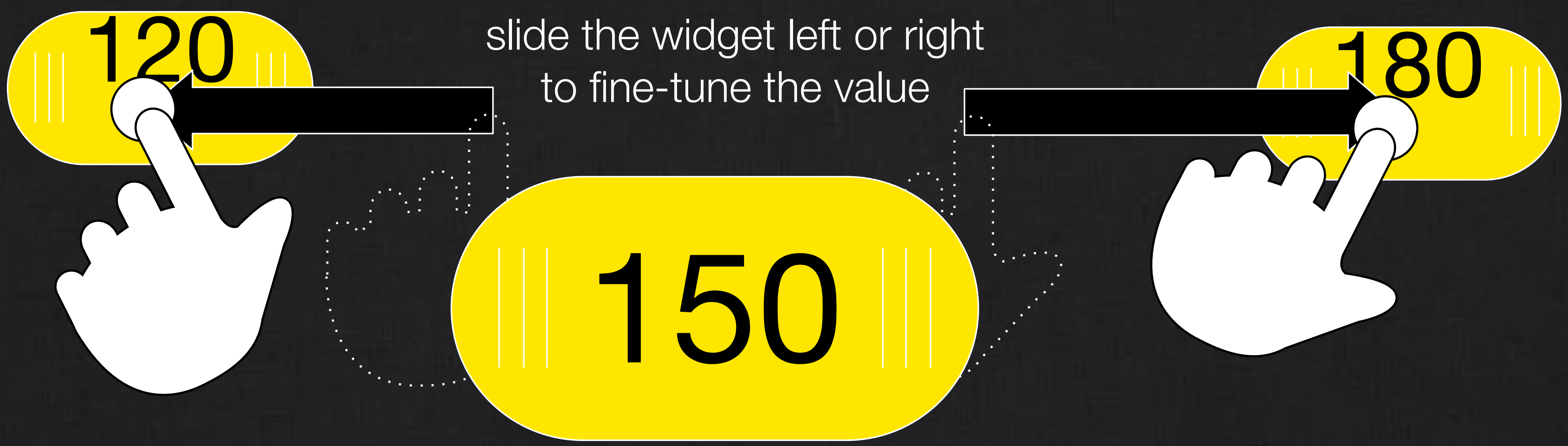
InterState editor





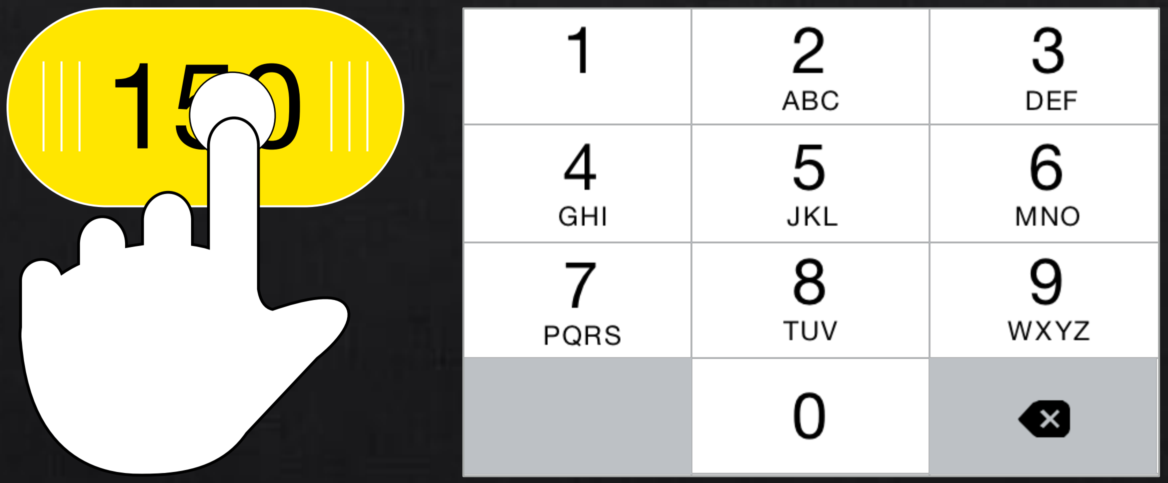
			hidden	dragging	showing
prototypes	(image	svg.image			
width	375	paper.width			
height	137	width*274/7			
x	0	0			
y	647	twoFingers.	paper.heigh	Math.max(pa	paper.heigh
href	'touch	'touch_imag			
show	true	true			
clip_rect	null	null			
cursor	'defau	'default'			
opacity	1	1.0			
transform	''	''			
animated_properties	false	false			
animation_duration	300	300			
animation_easing	'linea	'linear'			
shape	'image	'image'			

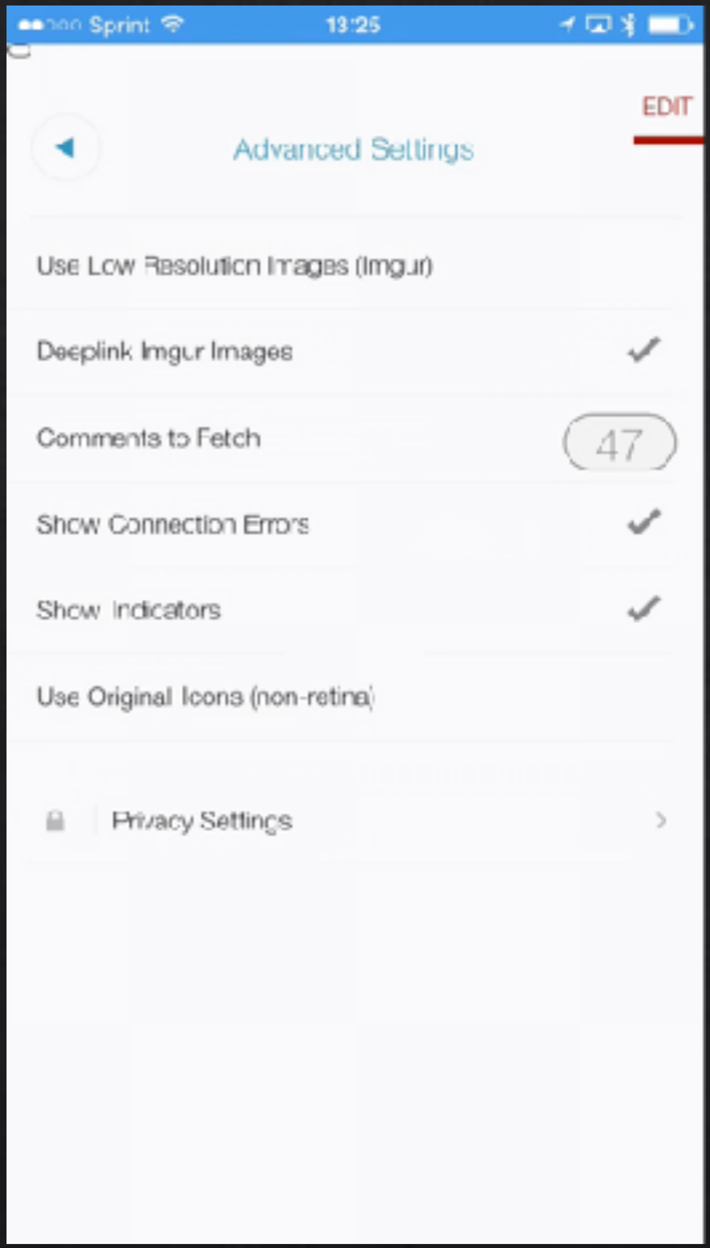
			hidden	dragging	showing
prototypes	(image	svg.image			
x	0	0			
y	647		paper.heigh	Math.max(pa	paper.heigh
width	375	colorsPanel			
height	137	colorsPanel			
href	'touch	'touch_imag			
show	true	true			
clip_rect	null	null			
cursor	'defau	'default'			
opacity	1	1.0			
transform	''	''			
animated_properties	false	false			
animation_duration	300	300			
animation_easing	'linea	'linear'			
shape	'image	'image'			



slide the widget left or right
to fine-tune the value

tap the widget to invoke
the numeric keyboard





Advanced Settings

Use Low Resolution Images (Imgur)

Deeplink Imgur Images ✓

Comments to Fetch 47

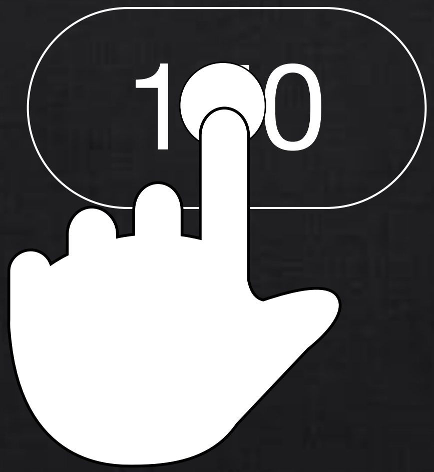
Show Connection Errors ✓

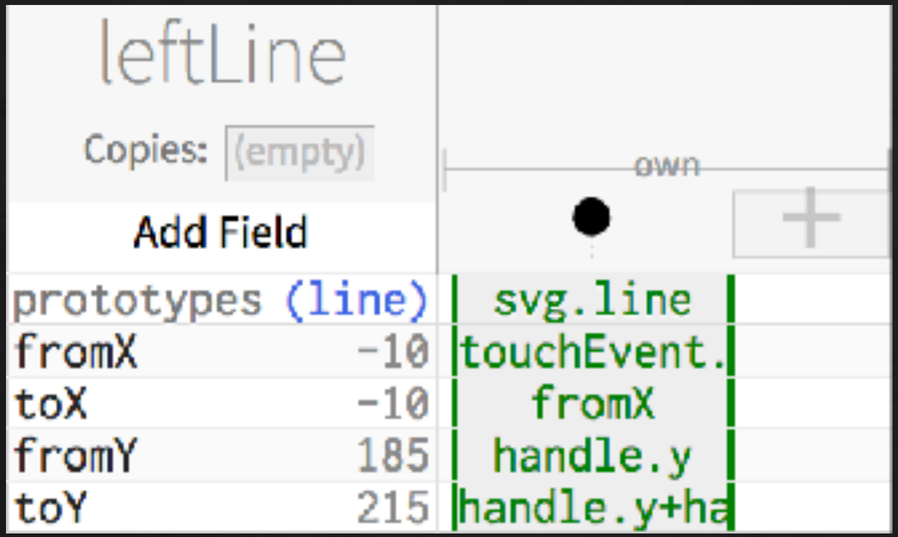
Show Indicators ✓

Use Original Icons (non-retina)

Privacy Settings ›

150

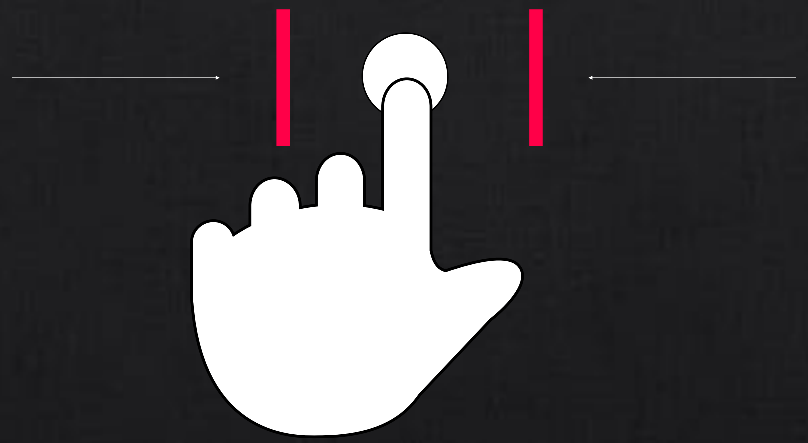

150

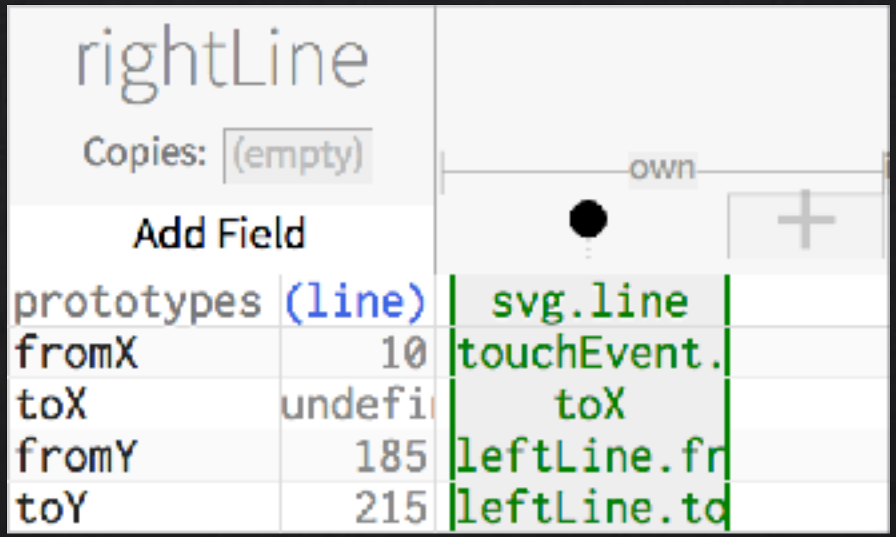


leftLine

Copies: (empty)

prototypes	(line)	svg.line
fromX	-10	touchEvent.
toX	-10	fromX
fromY	185	handle.y
toY	215	handle.y+ha

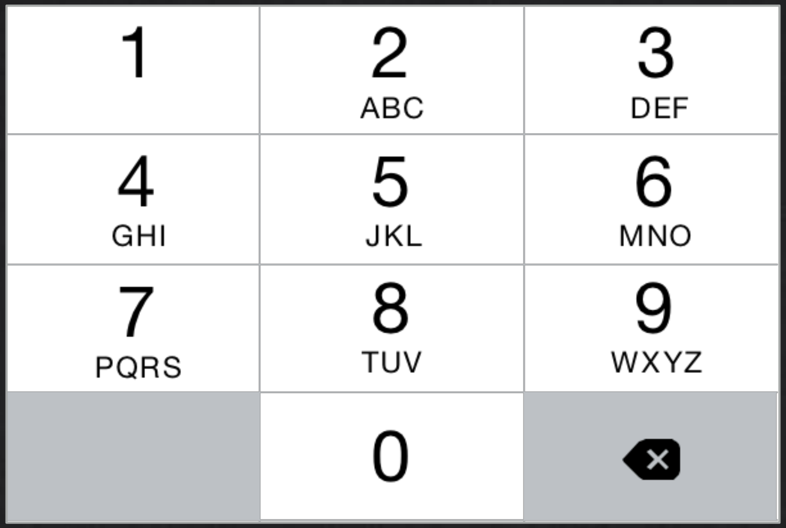




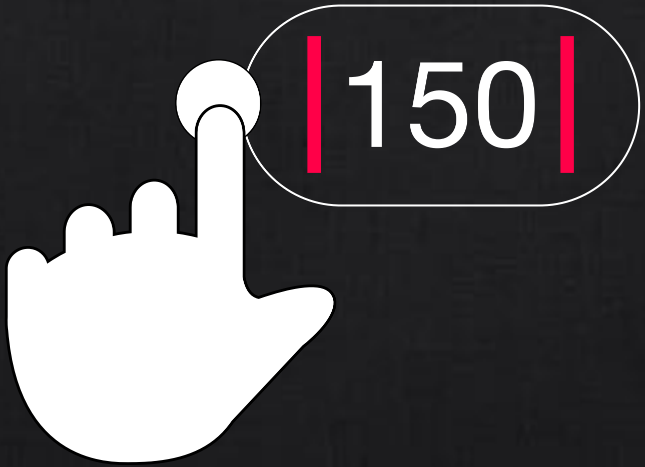
rightLine

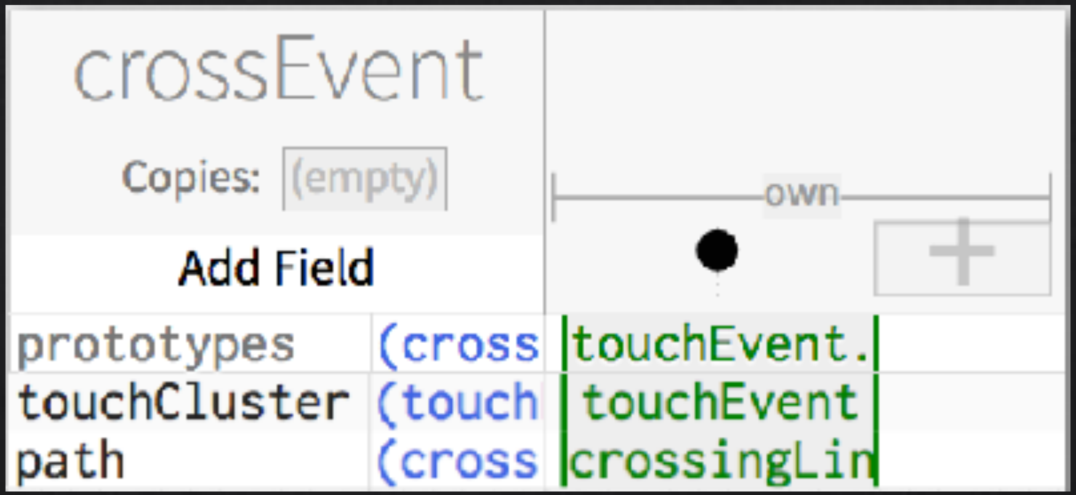
Copies: (empty)

prototypes	(line)	svg.line
fromX	10	touchEvent.
toX	undefi	toX
fromY	185	leftLine.fr
toY	215	leftLine.to



1
2 ABC
3 DEF
4 GHI
5 JKL
6 MNO
7 PQRS
8 TUV
9 WXYZ
0
⌫



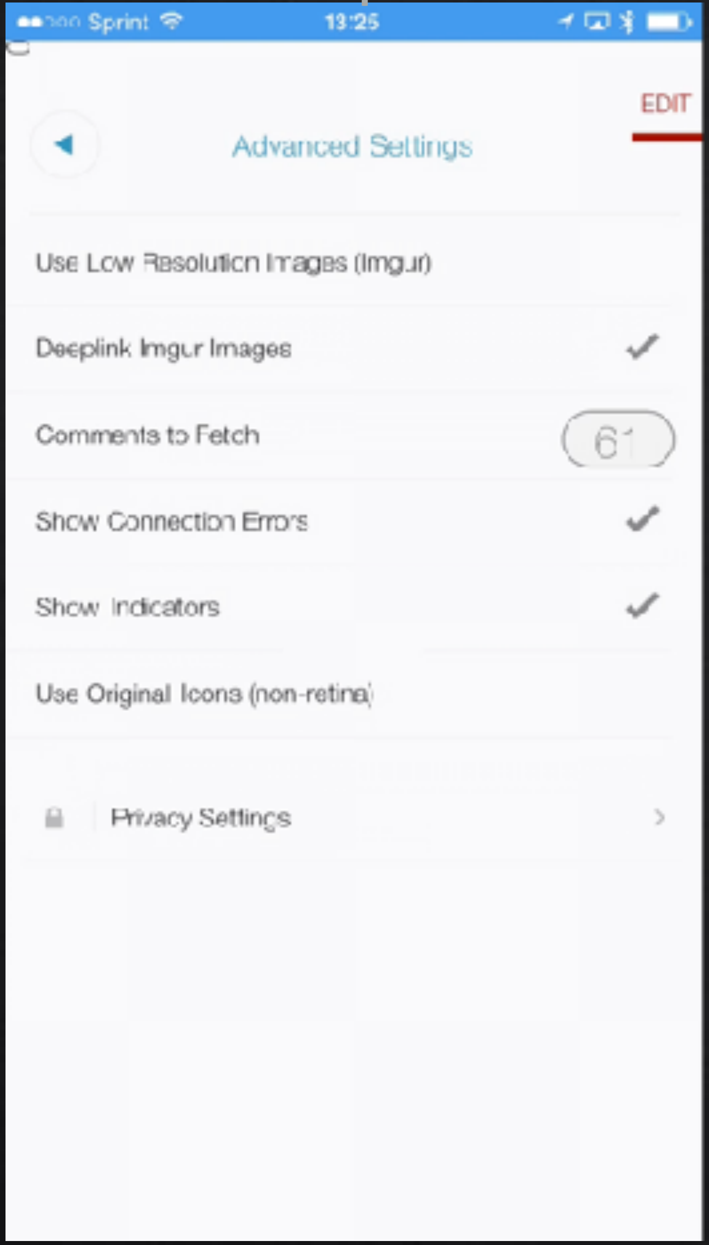

crossEvent

Copies: (empty)

Add Field

prototypes	(cross	touchEvent.
touchCluster	(touch	touchEvent
path	(cross	crossingLin

output



Sprint 13:25

EDIT

Advanced Settings

Use Low Resolution Images (Imgur)

Deeplink Imgur Images ✓

Comments to Fetch 61

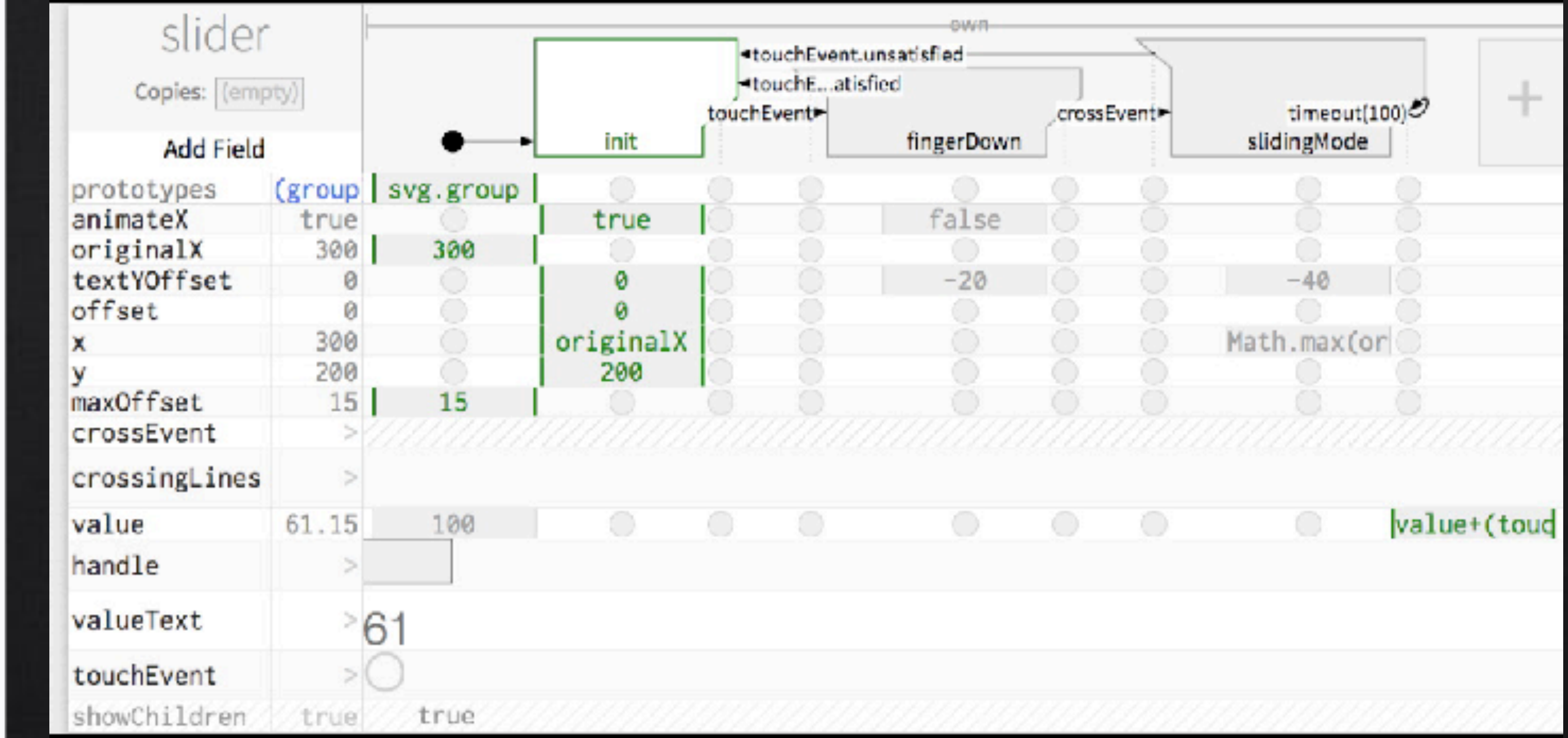
Show Connection Errors ✓

Show Indicators ✓

Use Original Icons (non-retina)

Privacy Settings ›

InterState editor

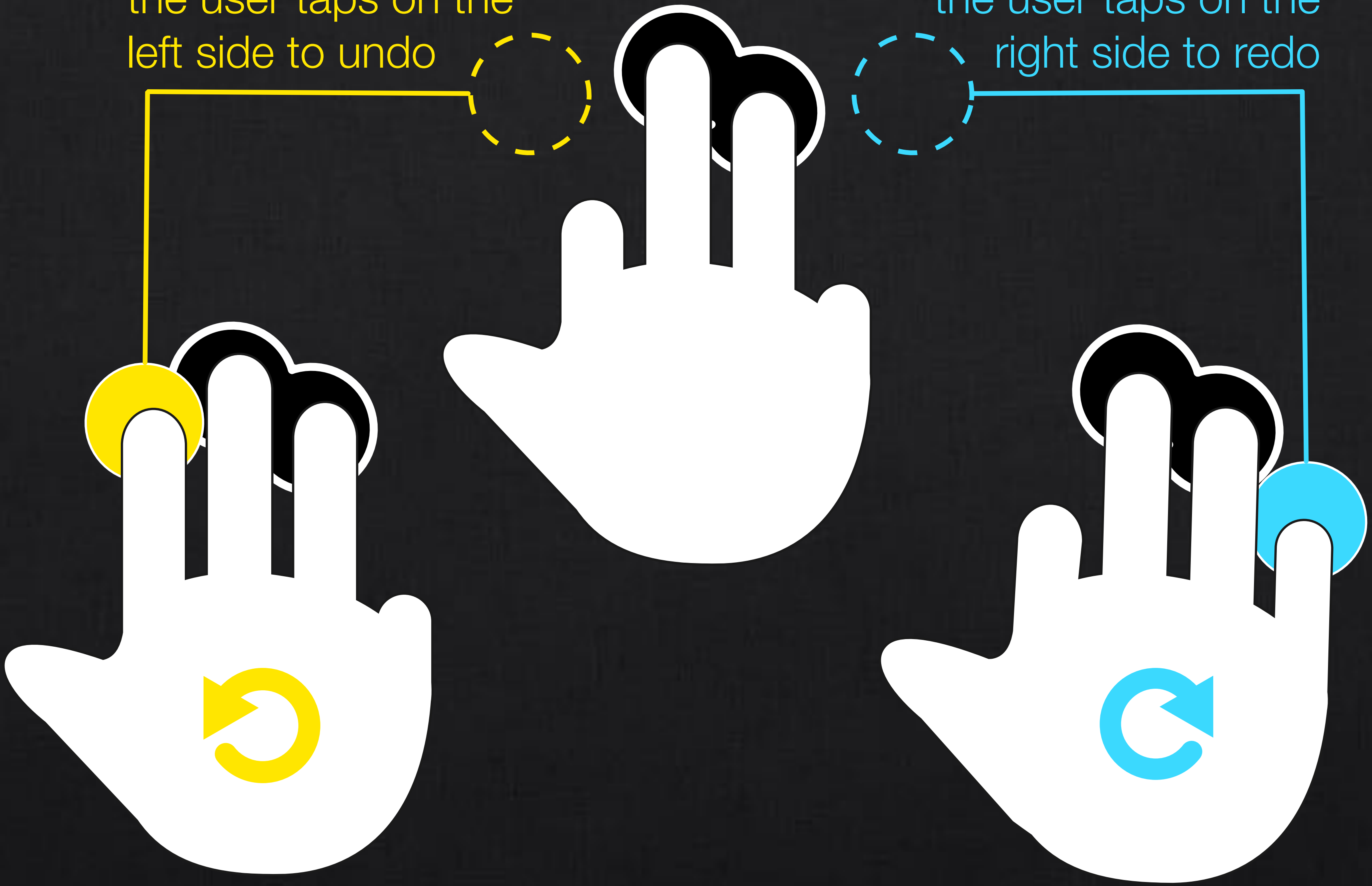


slider

Copies: (empty)

Add Field

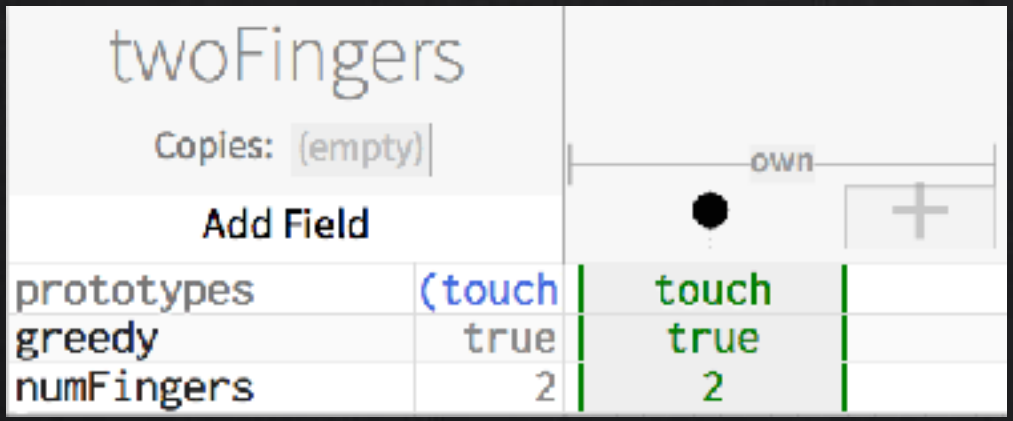
			init	fingerDown	slidingMode	
prototypes	(group	svg.group				
animateX	true		true	false		
originalX	300	300				
textYOffset	0		0	-20	-40	
offset	0		0			
x	300		originalX		Math.max(or	
y	200		200			
maxOffset	15	15				
crossEvent	>					
crossingLines	>					
value	61.15	100				value+(toud
handle	>					
valueText	>	61				
touchEvent	>	○				
showChildren	true	true				



the user presses two fingers down

the user taps on the left side to undo

the user taps on the right side to redo



twoFingers

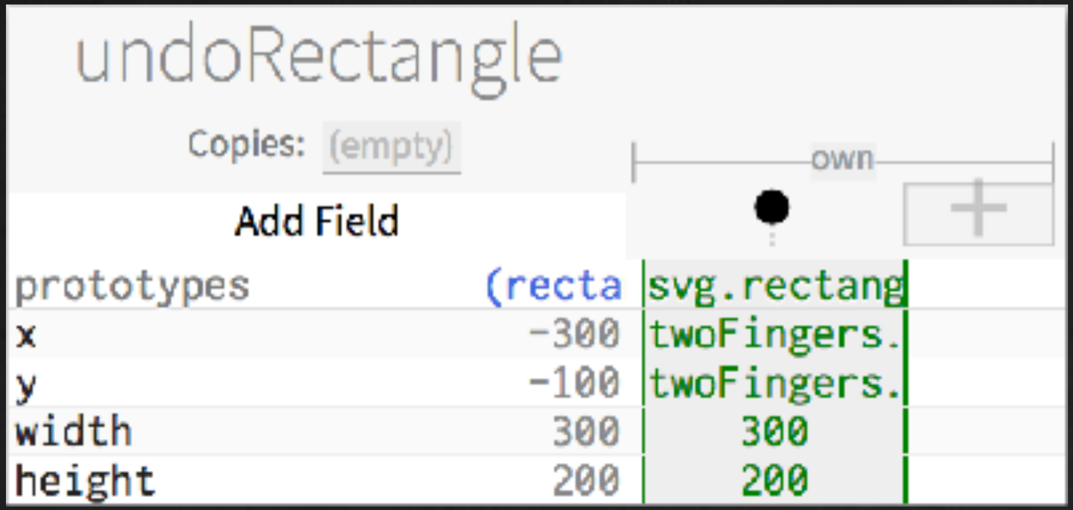
Copies: (empty)

Add Field

own

●

prototypes	(touch	touch	
greedy	true	true	
numFingers	2	2	



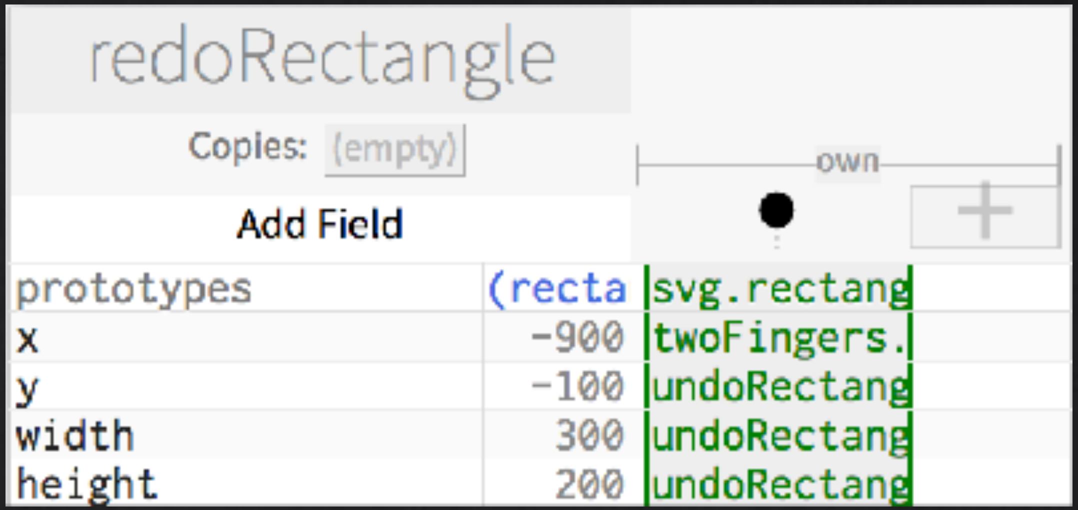
undoRectangle

Copies: (empty)

own

Add Field

prototypes	(recta	svg.rectang
x	-300	twoFingers.
y	-100	twoFingers.
width	300	300
height	200	200



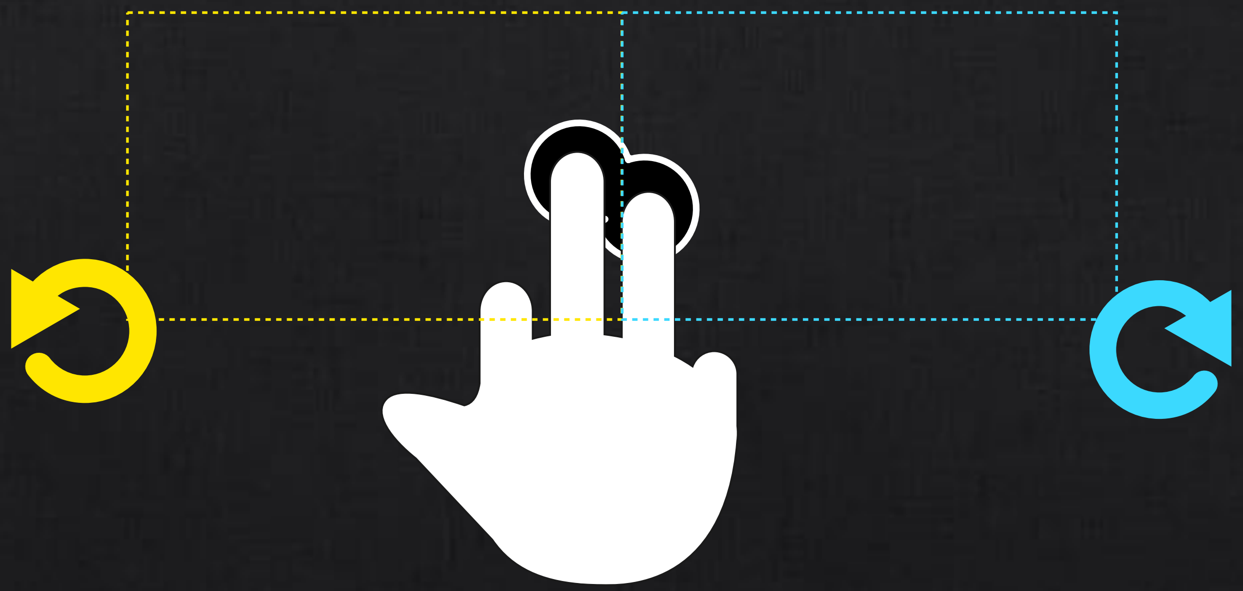
redoRectangle

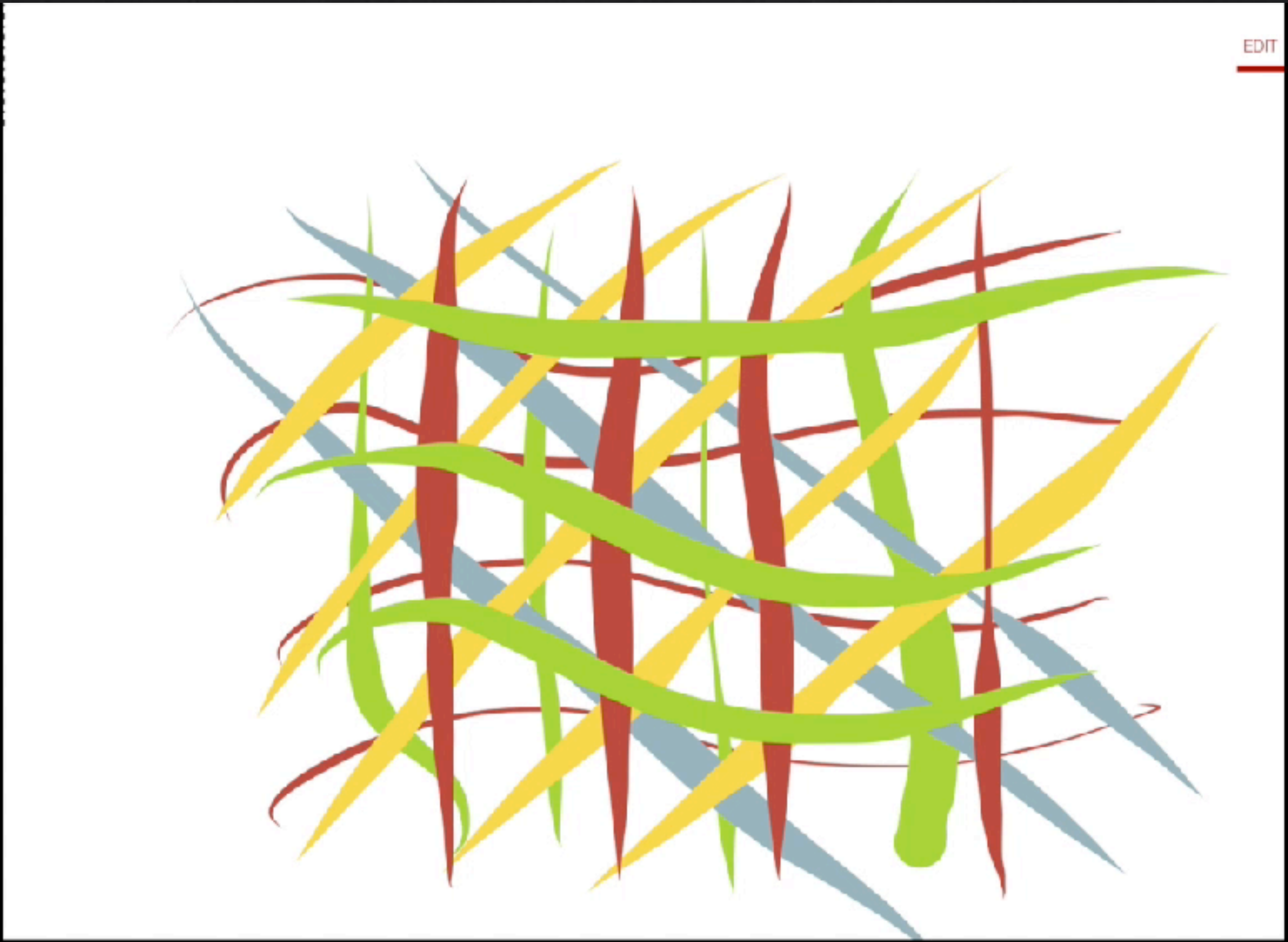
Copies: (empty)

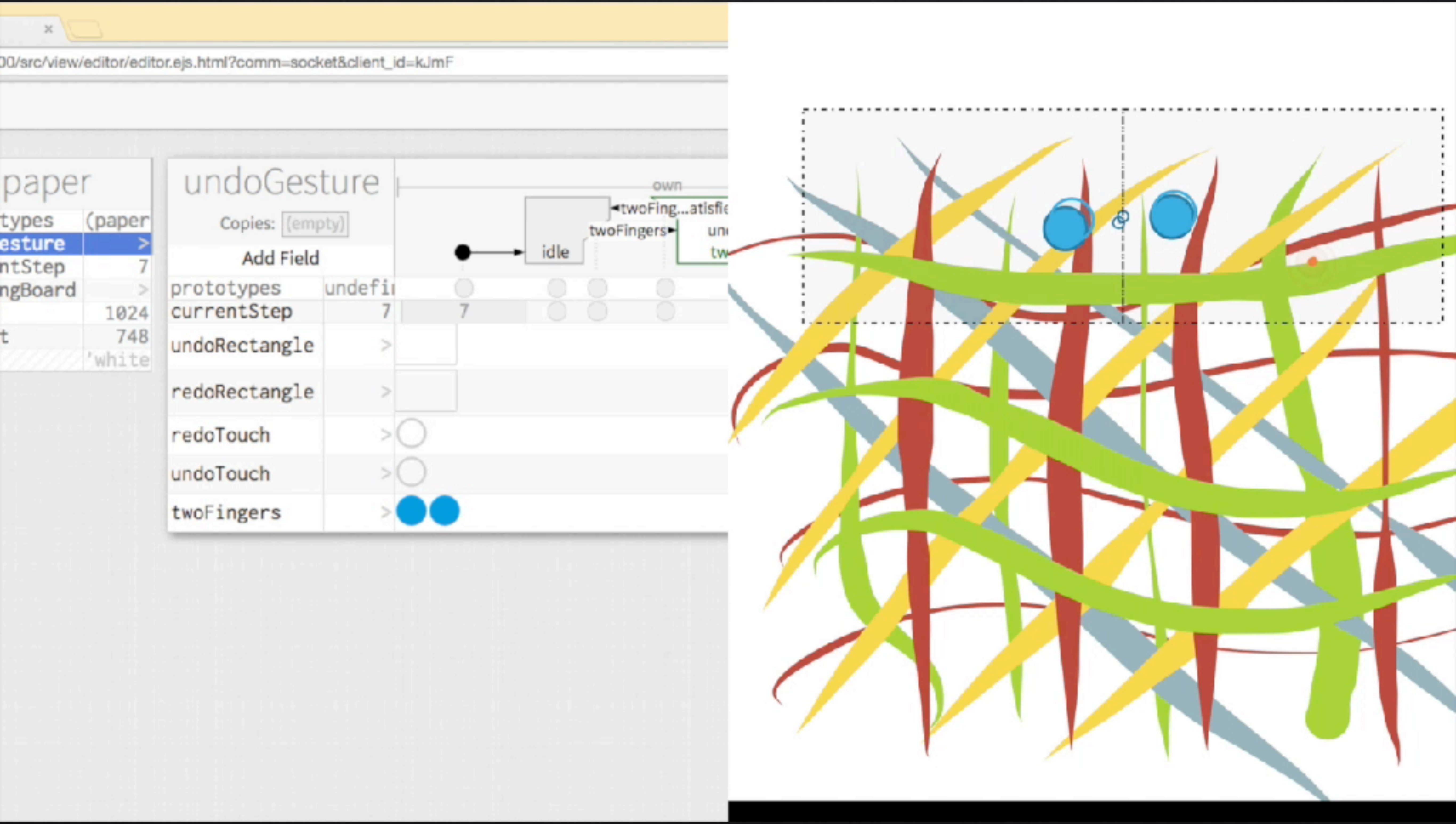
own

Add Field

prototypes	(recta	svg.rectang
x	-900	twoFingers.
y	-100	undoRectang
width	300	undoRectang
height	200	undoRectang




EDIT



00/src/view/editor/editor.ejs.html?comm=socket&client_id=kJmF

paper

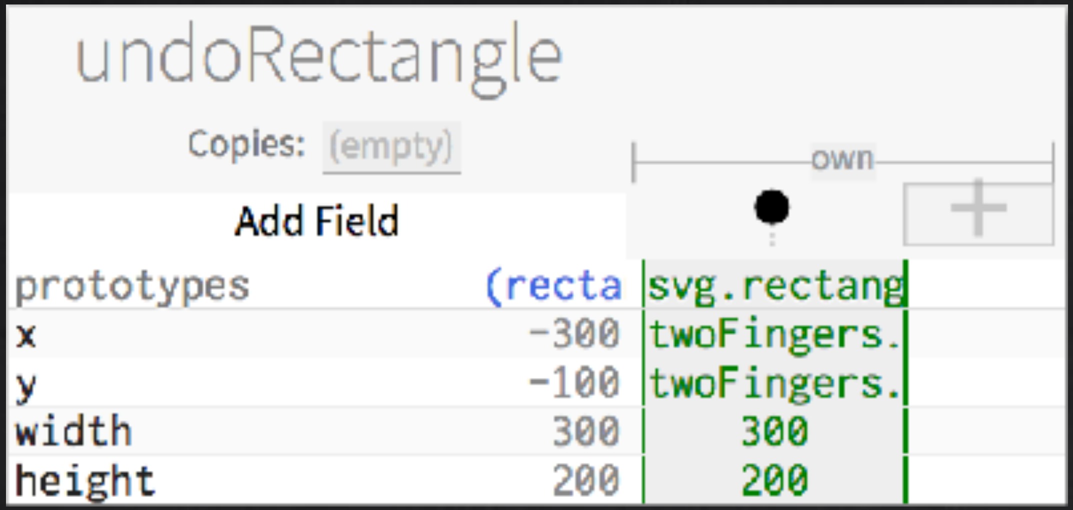
types	(paper
esture	>
ntStep	7
ngBoard	>
	1024
t	748
	'white

undoGesture

Copies: (empty)

Add Field

prototypes	undefi	
currentStep	7	7
undoRectangle	>	
redoRectangle	>	
redoTouch	>	○
undoTouch	>	○
twoFingers	>	● ●



undoRectangle

Copies: (empty)

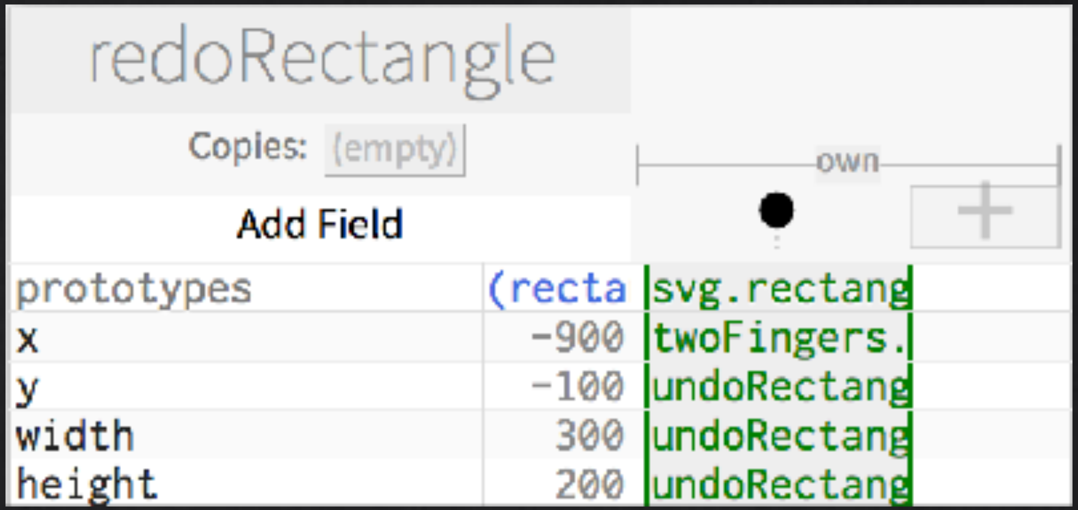
own

●

+

Add Field

prototypes	(recta	svg.rectang
x	-300	twoFingers.
y	-100	twoFingers.
width	300	300
height	200	200



redoRectangle

Copies: (empty)

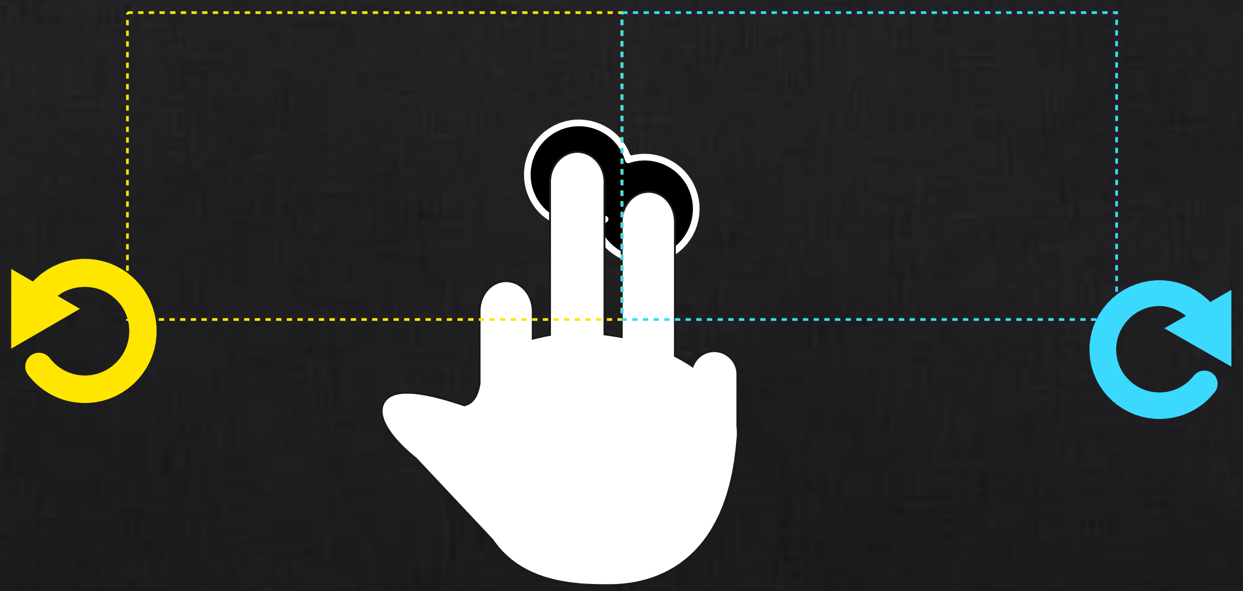
own

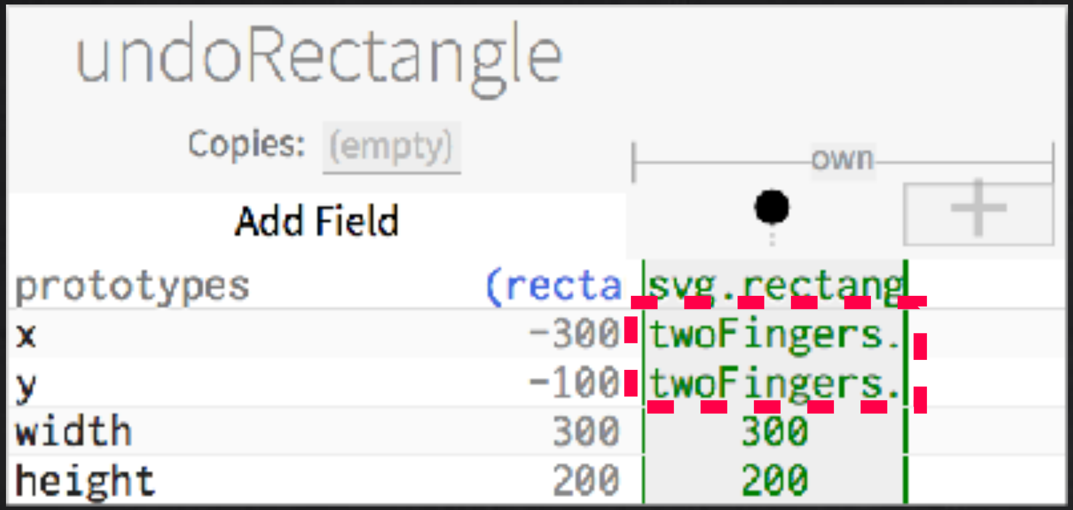
●

+

Add Field

prototypes	(recta	svg.rectang
x	-900	twoFingers.
y	-100	undoRectang
width	300	undoRectang
height	200	undoRectang





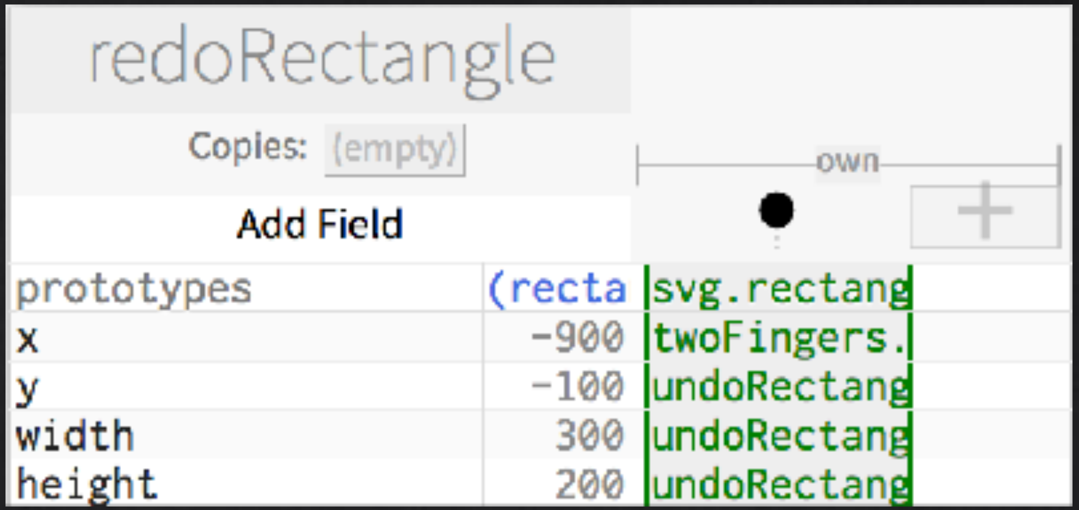
undoRectangle

Copies: (empty)

own

Add Field

prototypes	(recta	svg.rectang
x	-300	twoFingers.
y	-100	twoFingers.
width	300	300
height	200	200



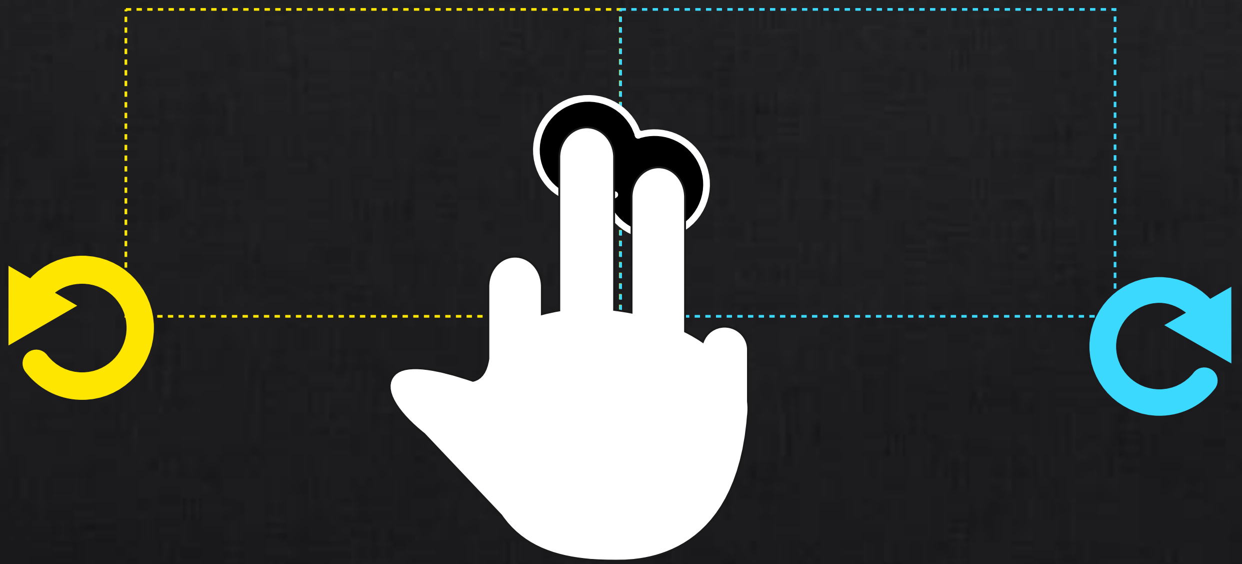
redoRectangle

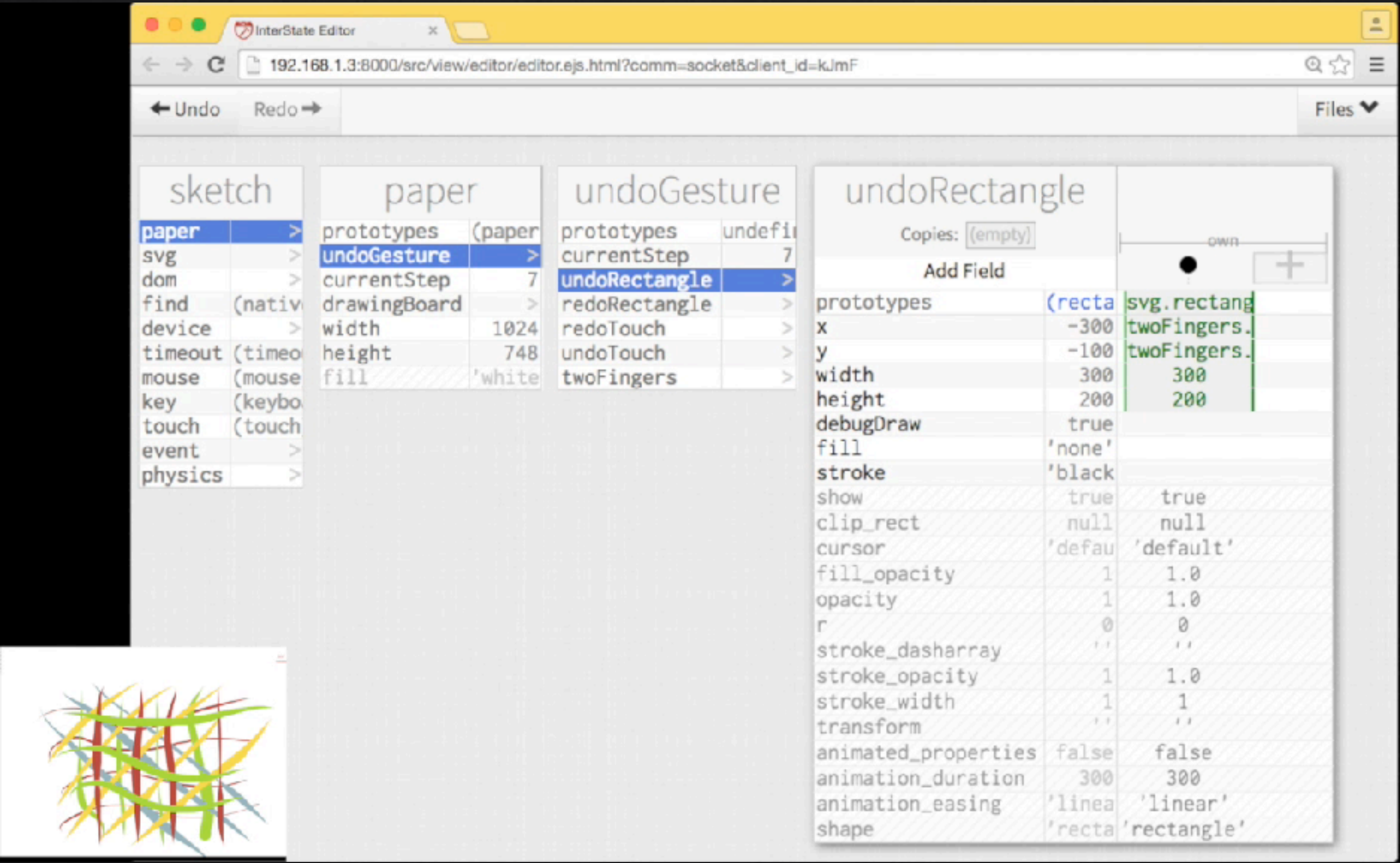
Copies: (empty)

own

Add Field

prototypes	(recta	svg.rectang
x	-900	twoFingers.
y	-100	undoRectang
width	300	undoRectang
height	200	undoRectang




InterState Editor
192.168.1.3:8000/src/view/editor/editor.ejs.html?comm=socket&client_id=kJmF
Undo
Redo
Files

sketch

paper	>
svg	>
dom	>
find	(nativ
device	>
timeout	(timeo
mouse	(mouse
key	(keybo
touch	(touch
event	>
physics	>

paper

prototypes	(paper
undoGesture	>
currentStep	7
drawingBoard	>
width	1024
height	748
fill	'white

undoGesture

prototypes	undefi
currentStep	7
undoRectangle	>
redoRectangle	>
redoTouch	>
undoTouch	>
twoFingers	>

undoRectangle

Copies: (empty)

Add Field

own ● +

prototypes	(recta	svg.rectang
x	-300	twoFingers.
y	-100	twoFingers.
width	300	300
height	200	200
debugDraw	true	
fill	'none'	
stroke	'black	
show	true	true
clip_rect	null	null
cursor	'defau	'default'
fill_opacity	1	1.0
opacity	1	1.0
r	0	0
stroke_dasharray	''	''
stroke_opacity	1	1.0
stroke_width	1	1
transform	''	''
animated_properties	false	false
animation_duration	300	300
animation_easing	'linea	'linear'
shape	'recta	'rectangle'

creative tool integration

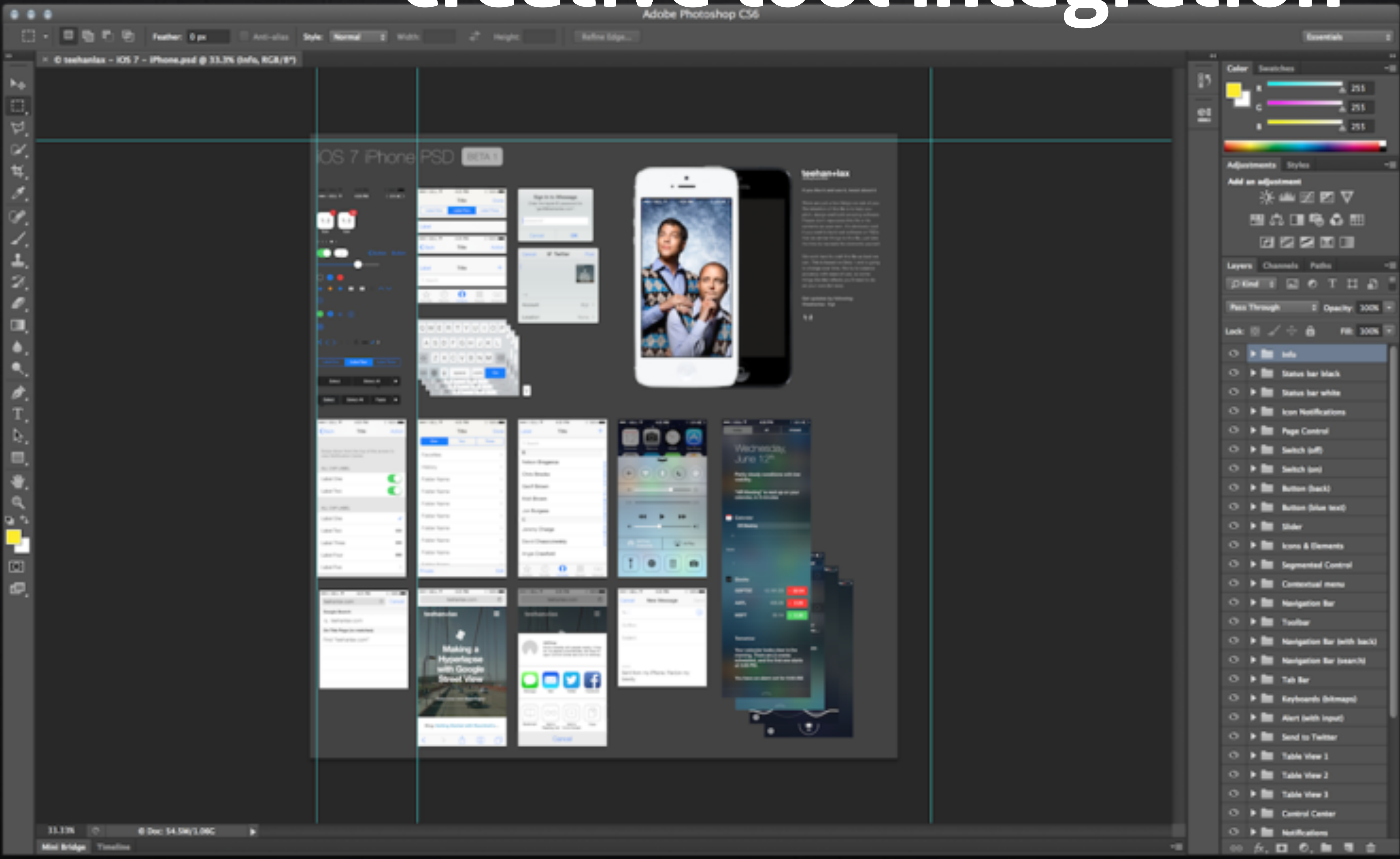


photo: iMore

contributions

- constraint model integrating state
- library for developers (ConstraintJS)
- visual notation of constraint model
- live editor for visual notation
- primitives for inheritance & templating
- evaluation of model, primitives & visual notation
- multi-touch development primitives

thank you

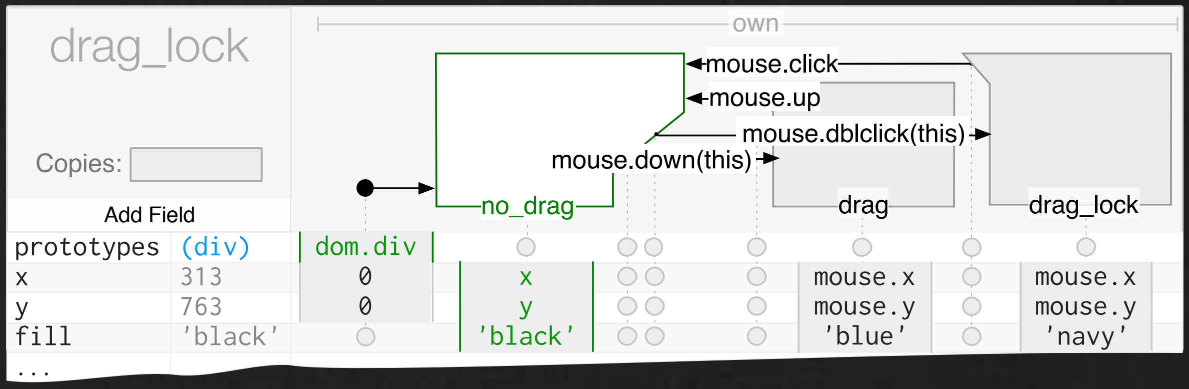
advisors
Brad Myers and Joel Brandt

committee
Scott Hudson and John Zimmerman

funding
Adobe, ARCS, Google/UNCF, Ford Foundation, Microsoft, NSF

friends & family
Justine Cassell, Kerry Chang, Chris Harrison, Iris Howley, Queenie Kravitz, Min Kyung Lee, Ian Li, Kurt Luther, Jennifer Marlow, Amy Ogan, Christina Oney, Logan Oney III, Theresa Oney, Julia Schwarz, Yla Tausczik, Eliane Wiese, Jason Wiese, Tao Xie, YoungSeok Yoon, and more people than I can fit into one slide.

```
friends = cjs.async(fb_request("/me/friends"));
pics    = friends.map(function(friend) {
              return cjs.async(fb_request( "/" + friend.id
                                          + "/picture"));
          });

//...

{{#diagram friends.state}}
   {{#state pending }} Loading friends...
   {{#state rejected}} Error
   {{#state resolved}}
      {{#each friends friend i}}
         {{#diagram pics[i].state}}
            {{#state pending }} <img src = "loading.gif" />
            {{#state resolved}} <img src = "{{pics[i]}}" />
            {{#state rejected}} <img src = "error.gif"   />
         {{/diagram}}
         {{friend.name}}
      {{/each}}
{{/diagram}}
```



			no_drag	drag	drag_lock
prototypes	(div)	dom.div			
x	313	0	x	mouse.x	mouse.x
y	763	0	y	mouse.y	mouse.y
fill	'black'		'black'	'blue'	'navy'
...					

ConstraintJS

http://cjs.from.so

InterState

http://interstate.from.so

Stephen Oney / Expressing Interactivity with States and Constraints

http://from.so
soney@umich.edu

Funding for this research comes from a Microsoft SEIF award, multiple grants from Adobe, an ARCS Scholarship, a Ford Foundation Fellowship, and from NSF grant IIS-1116724.