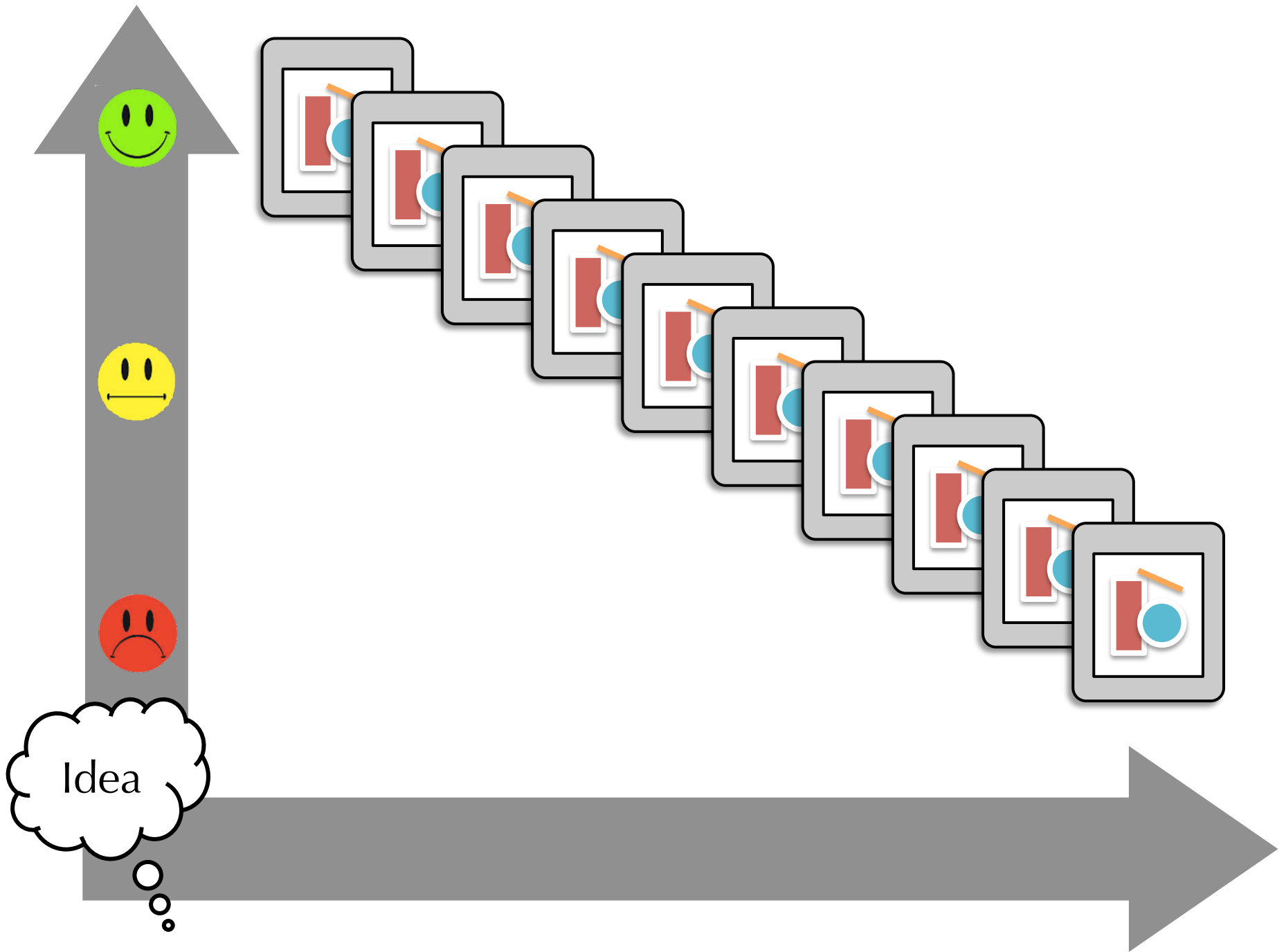
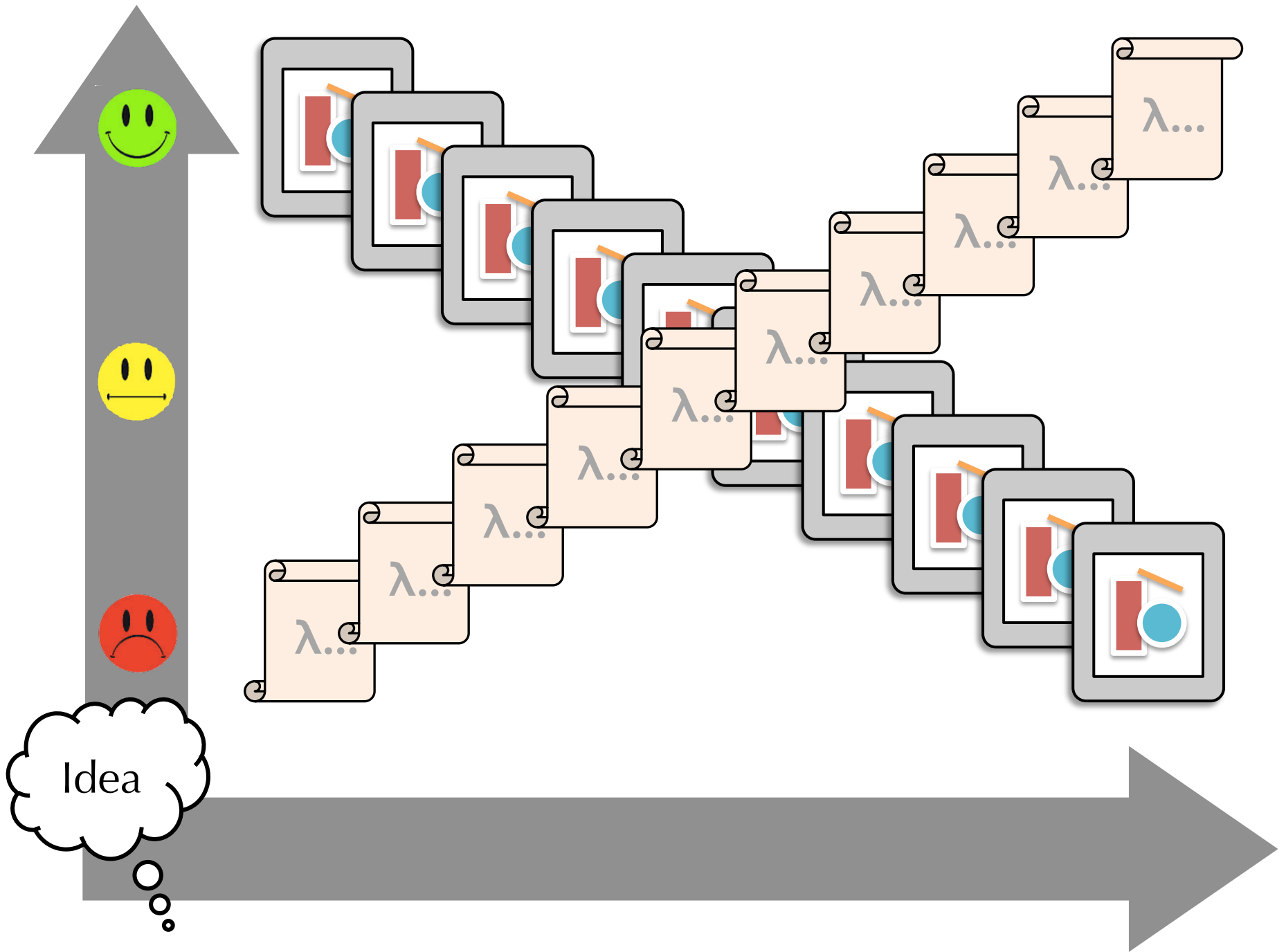
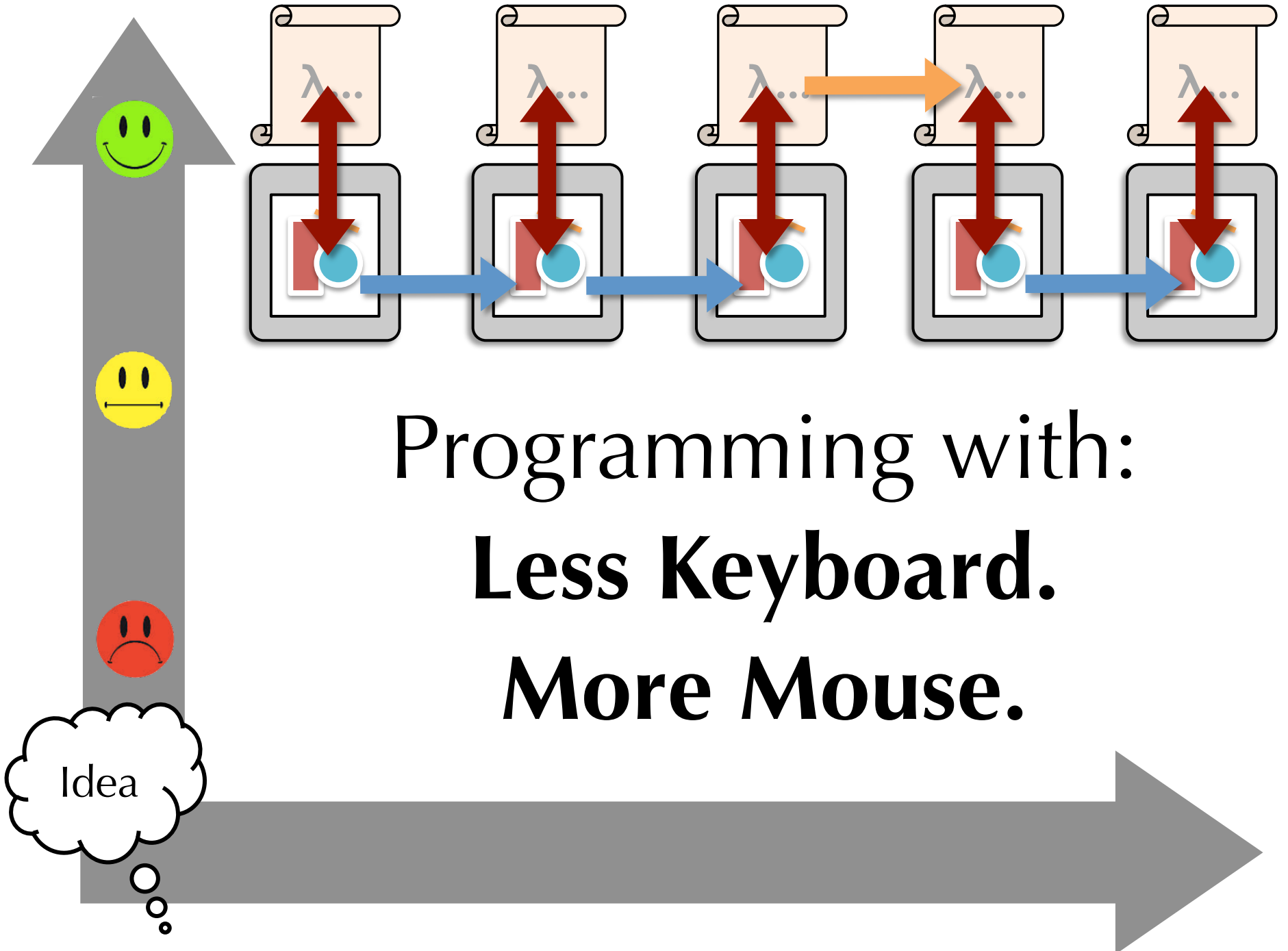









Topics in PL:
Direct Manipulation
Programming







Programming with:
Less Keyboard.
More Mouse.

	Programming Language	Direct Manipulation
2D Graphics		 
Presentations		 
Documents		 
Spreadsheets		 
Web Apps		 

General-Purpose Languages
















Domain-Specific Languages or Libraries

Domain-Specific GUI Tools

Language

Manipulation

2D Graphics			 
Presentations		 	 
Documents		 	 
Spreadsheets			 
Web Apps		 	 

	Programming Language	Direct Manipulation	Direct Manipulation Programming
2D Graphics	JS 	Ai 	 Sketch -n- Sketch
Presentations	JS 	 	
Documents	JS 	 	
Spreadsheets	JS 	 	
Web Apps	JS 	 	

λ-Calculus + Direct Manipulation

0 : Text-Edit Code

```
Sketch-n-Sketch  File  Code Tools  Output Tools  View  Options
Run ▶
...okeLen rotAngle)
4 (let center [(circle 'black' cx cy rCenter)]
5 (let frame [(nStar 'goldenrod' 'darkgray' 3 numSpokes spokeLen)
6 (let spokePts (nPointsOnCircle numSpokes rotAngle cx cy spokeLen)
7 (let cars (mapi (\[i [x y]] (squareByCenter (if (= i 0) 'pink'
8 (let hubcaps (map (\[x y] (circle 'black' x y rCap)) spokePts)
9 (concat [rim cars center frame hubcaps])
10 )))))))
11
12 (def [cx cy spokeLen rCenter wCar rCap] [228 241 178 20 30 7])
13 (def [spokes angle] [14!{3-25} 0.062!{-3.14-3.14}])
14
15 (svg (Cap spokes spokeLen angle))
Code Tools
Introduce Variables
Make Equal with Single Variable
New variable: num
New variable: angle
New variable: spokes
New variable: spokes_angle
```

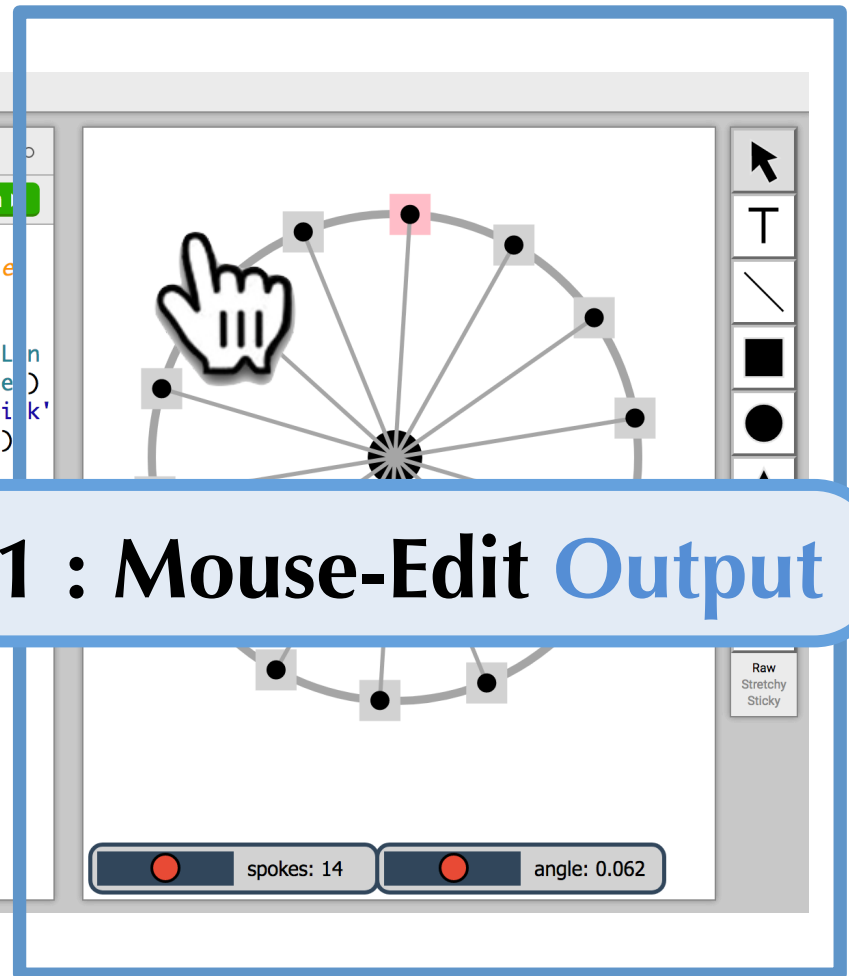
spokes: 14 angle: 0.062

λ -Calculus + Direct Manipulation of Output

0 : Text-Edit Code

```
Sketch-n-Sketch  File  Code Tools  Output Tools  View  Options  
Run  
4 (let center [(circle 'black' cx cy rCenter)]  
5 (let frame [(nStar 'goldenrod' 'darkgray' 3 numSpokes spokeLen  
6 (let spokePts (nPointsOnCircle numSpokes rotAngle cx cy spokeLen)  
7 (let cars (mapi (\[i [x y]] (squareByCenter (if (= i 0) 'pink'  
8 (let hubcaps (map (\[x y] (circle 'black' x y rCap)) spokePts)  
9 (concat [rim cars center frame hubcaps])  
10 )))))))  
11  
12 (def [cx cy spokeLen rCenter wCar rCap] [228 241 178 20 30  
13 (def [spokes angle] [14!{3-25} 0.062!{-3.14-3.14}])  
14  
15 (svg (Cap spokes spokeLen angle))  
Code Tools  
Introduce Variables  
Make Equal with Single Variable  
New variable: num  
New variable: angle  
New variable: spokes  
New variable: spokes_angle
```

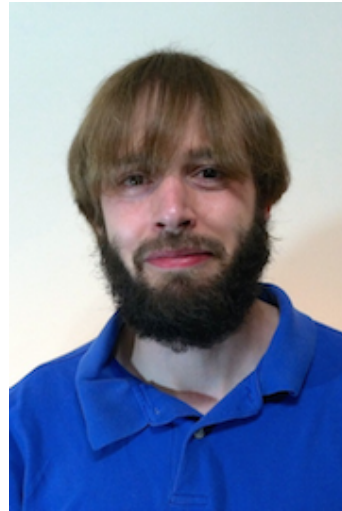
1 : Mouse-Edit Output



Sketch-n-Sketch: Direct Manipulation Programming



**Ravi
Chugh**



**Brian
Hempel**

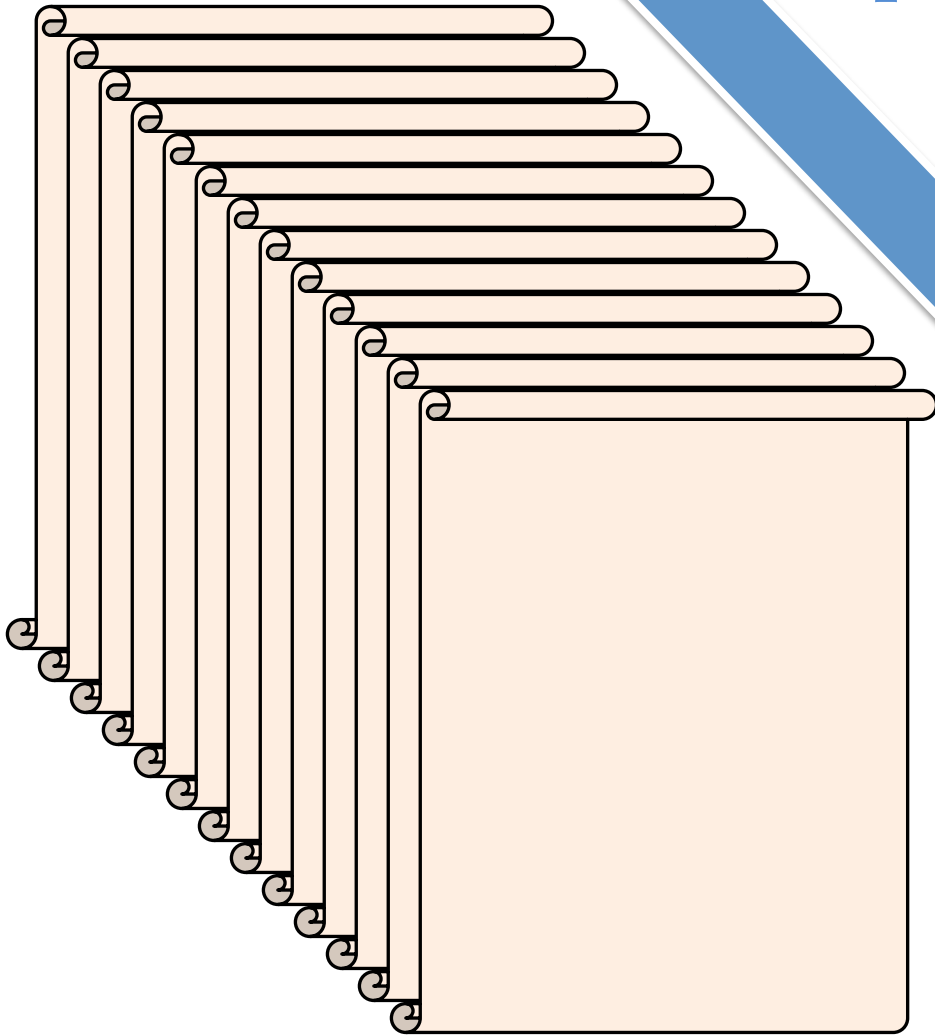


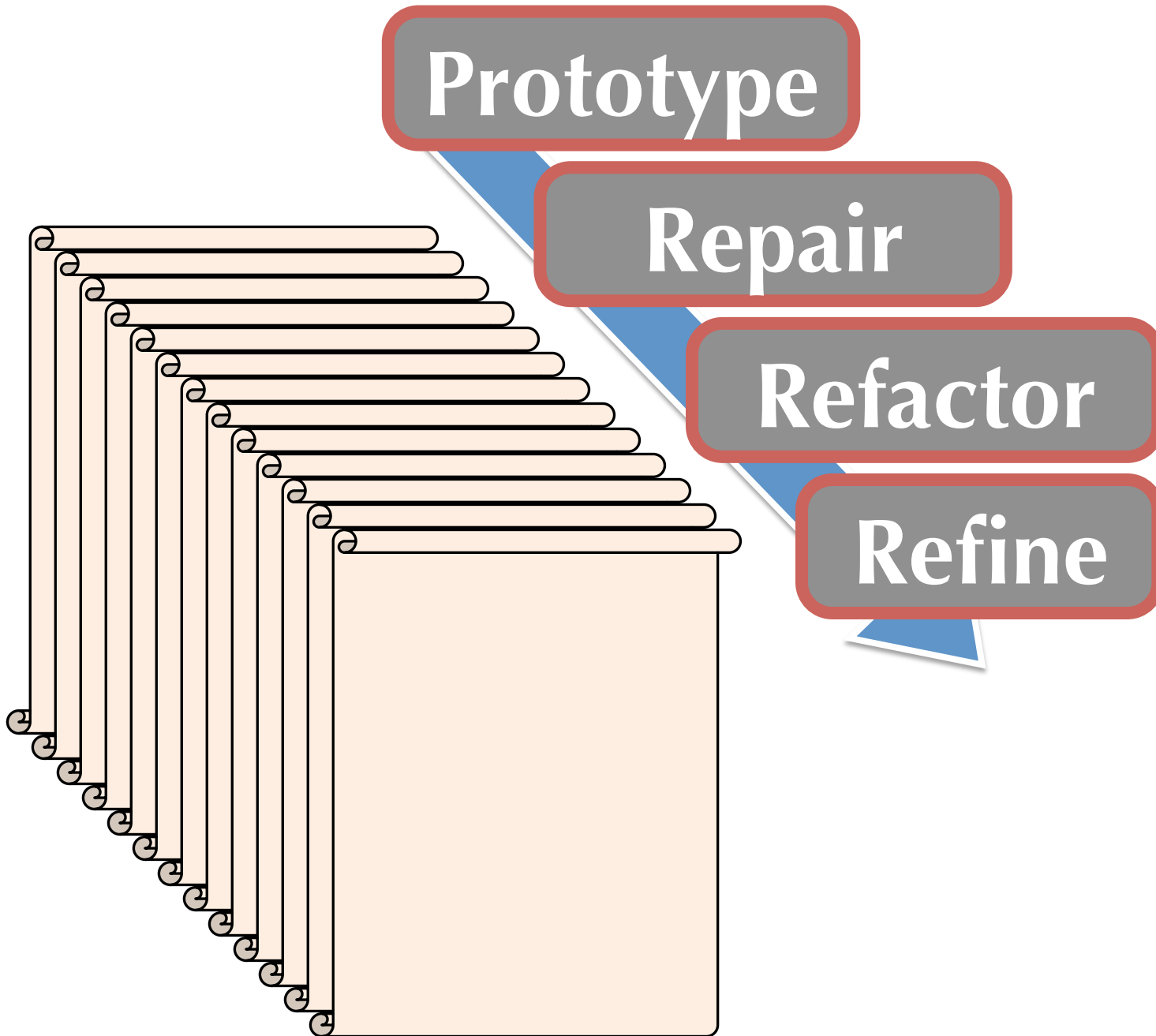
**Justin
Lubin**

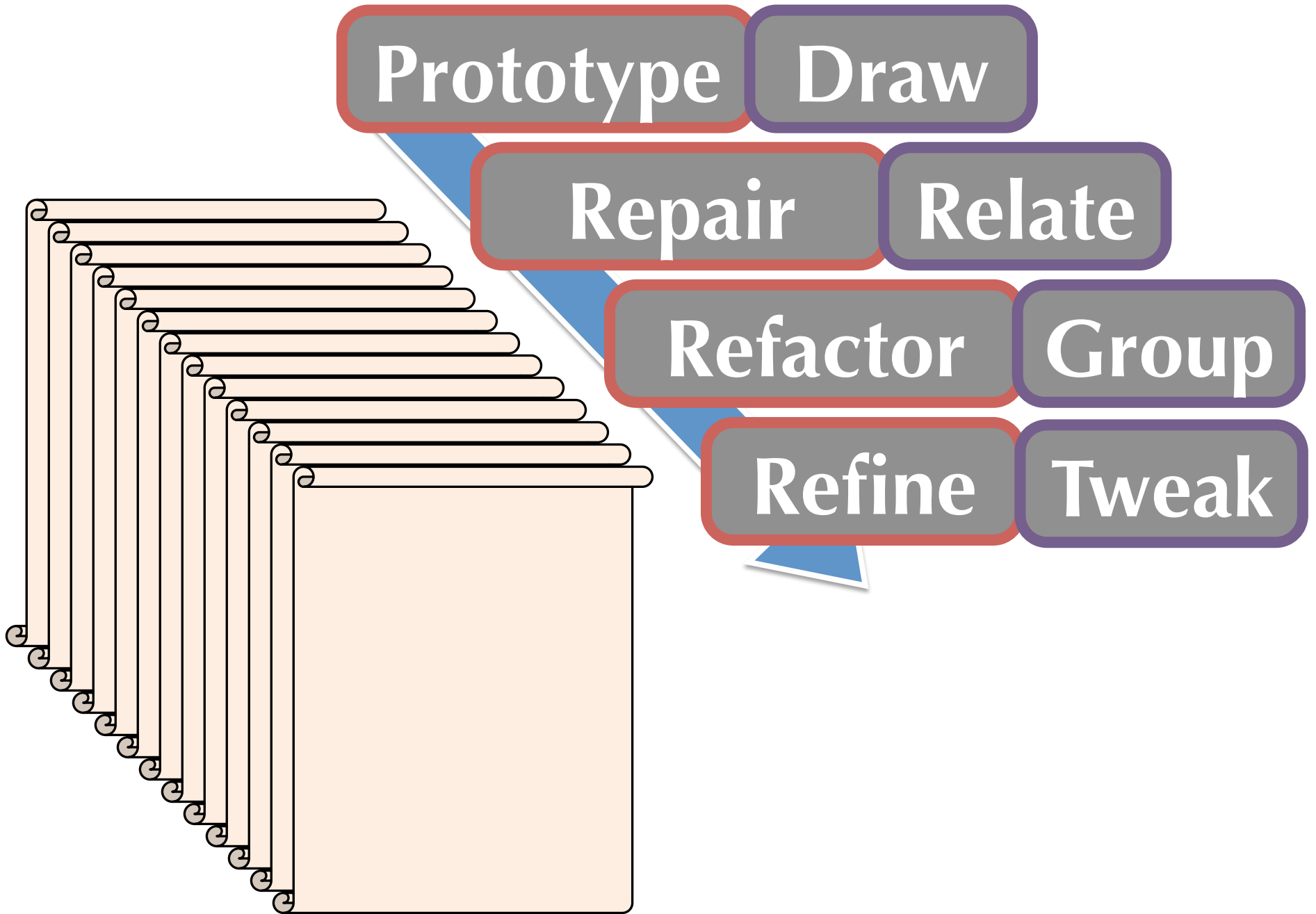


**Mikaël
Mayer**

Semi-Automated Programming by Manipulating Output







This Seminar:

**Brainstorming for Direct
Manipulation Programming**

MAKE

things with
Programming Languages
and/or
Direct Manipulation Editors

General-Purpose Languages

Domain-Specific Languages or Libraries

Domain-Specific GUI Tools

2D Graphics	JS		 	 Sketch -n- Sketch
Presentations	JS	 	 	
Documents	JS	 	 	
Spreadsheets	JS		 	
Web Apps	JS	 	 	

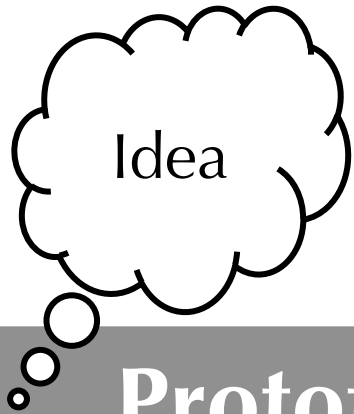
This Seminar:
**Brainstorming for Direct
Manipulation Programming**

MAKE

things with
Programming Languages
and/or
Direct Manipulation Editors

READ

papers about
Programming Languages
and/or
Direct Manipulation Editor



Prototype; Repair; Refactor; Repeat



