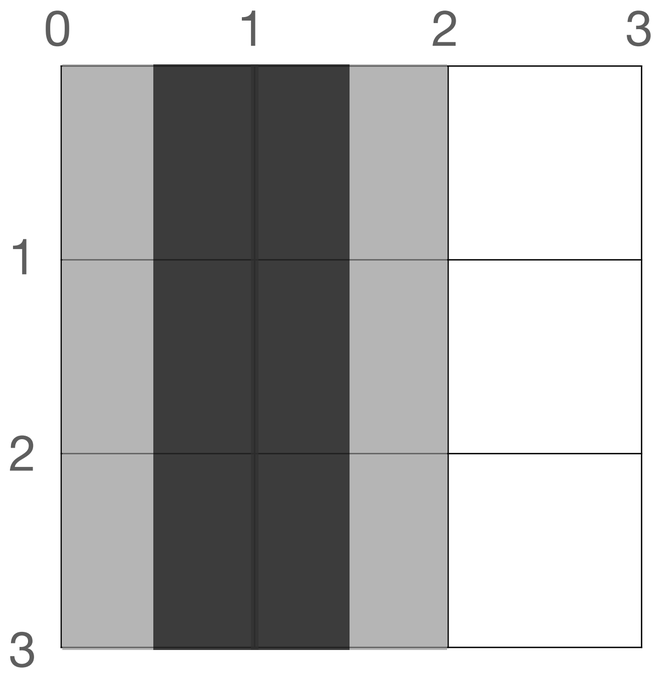
Javascript and Canvas

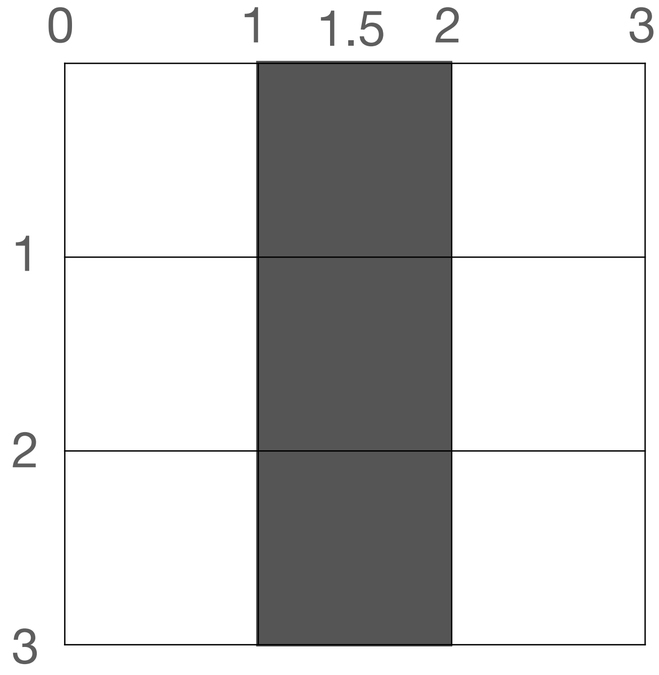
Dev Environment	IO	Flow control			
Create/Size/Reset Canvas	Rectangle Line, Arc LinearGradient	Text			
Basic types	Function	Composite types: Object / Array	Events		
Animation game loop					
OOP part 1	Audio				
Interaction	Collision detection	Images	Spritesheet	Transparent Shadow RadialGradient	Video
Transform	Projects: Tetris, 2D Breakout, ……			Composite	

Why line blurry?



ctx.moveTo(1,0)
ctx.lineTo(1,3)
ctx.lineWidth=1

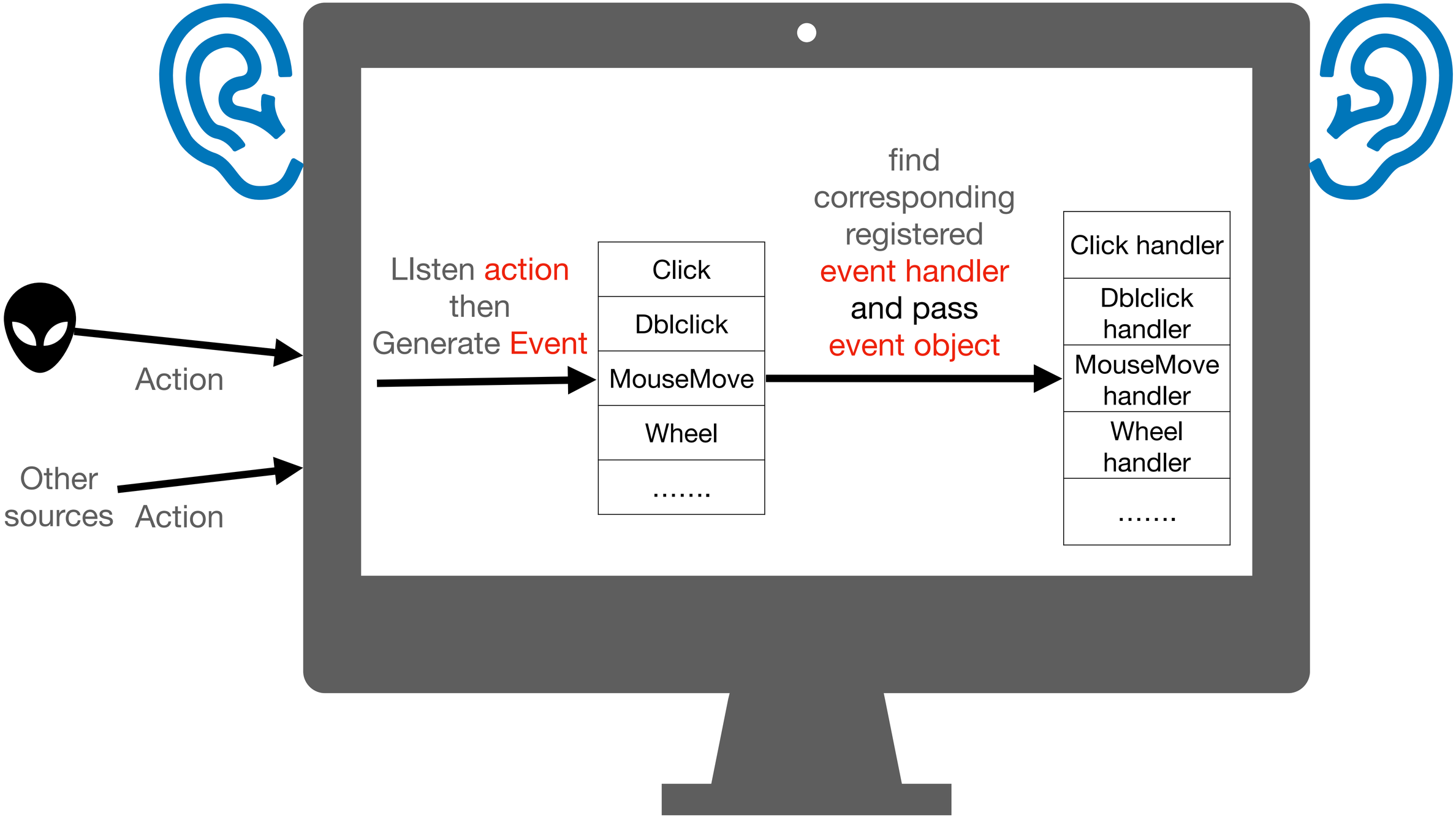
browser cannot draw half pixel



ctx.moveTo(1.5,0)
ctx.lineTo(1.5,3)
ctx.lineWidth=1

browser is able to draw a true 1 pixel

Events


Action
Other sources
Action
LIsten action then Generate Event
Click
Dblclick
MouseMove
Wheel
.....
find corresponding registered event handler and pass event object
Click handler
Dblclick handler
MouseMove handler
Wheel handler
.....

Arrow keys

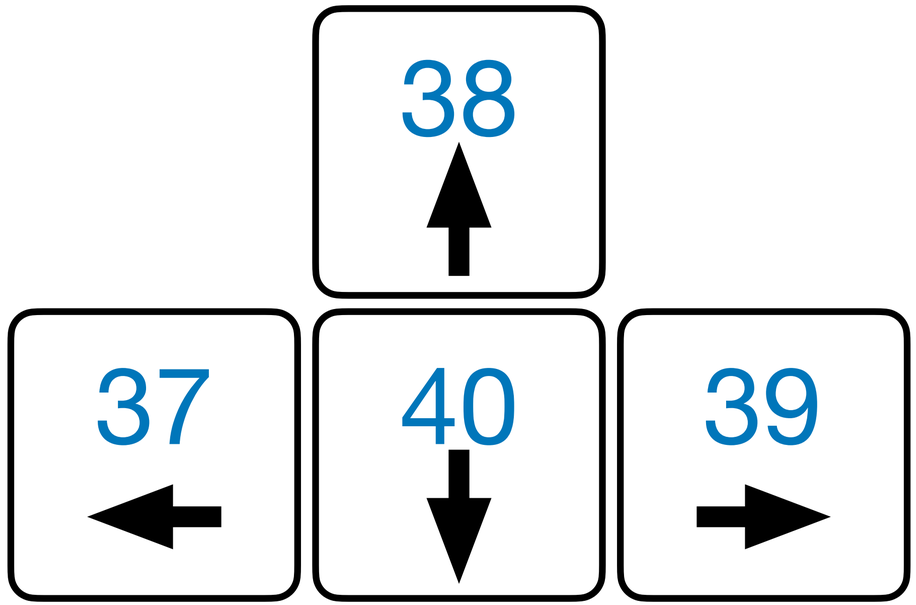
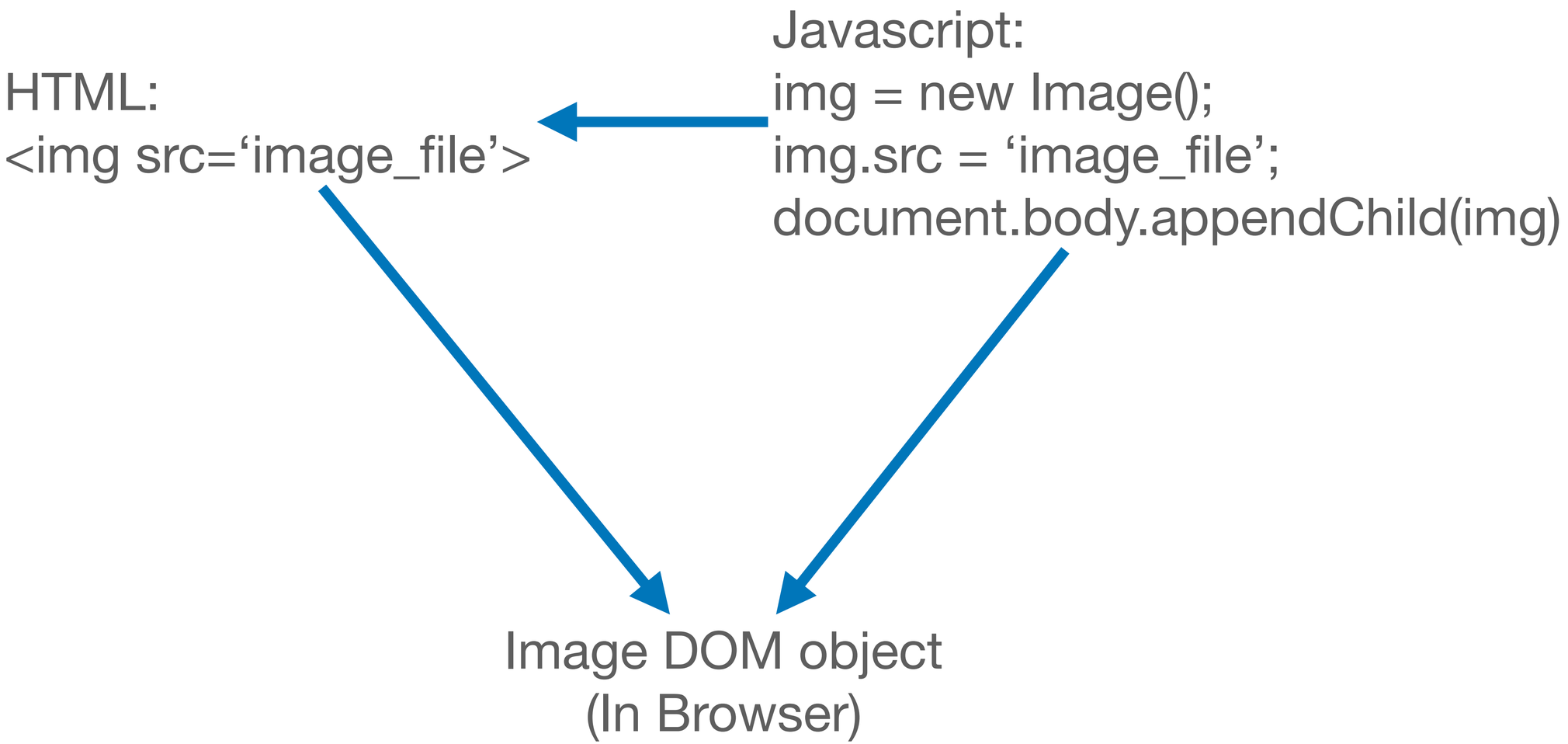


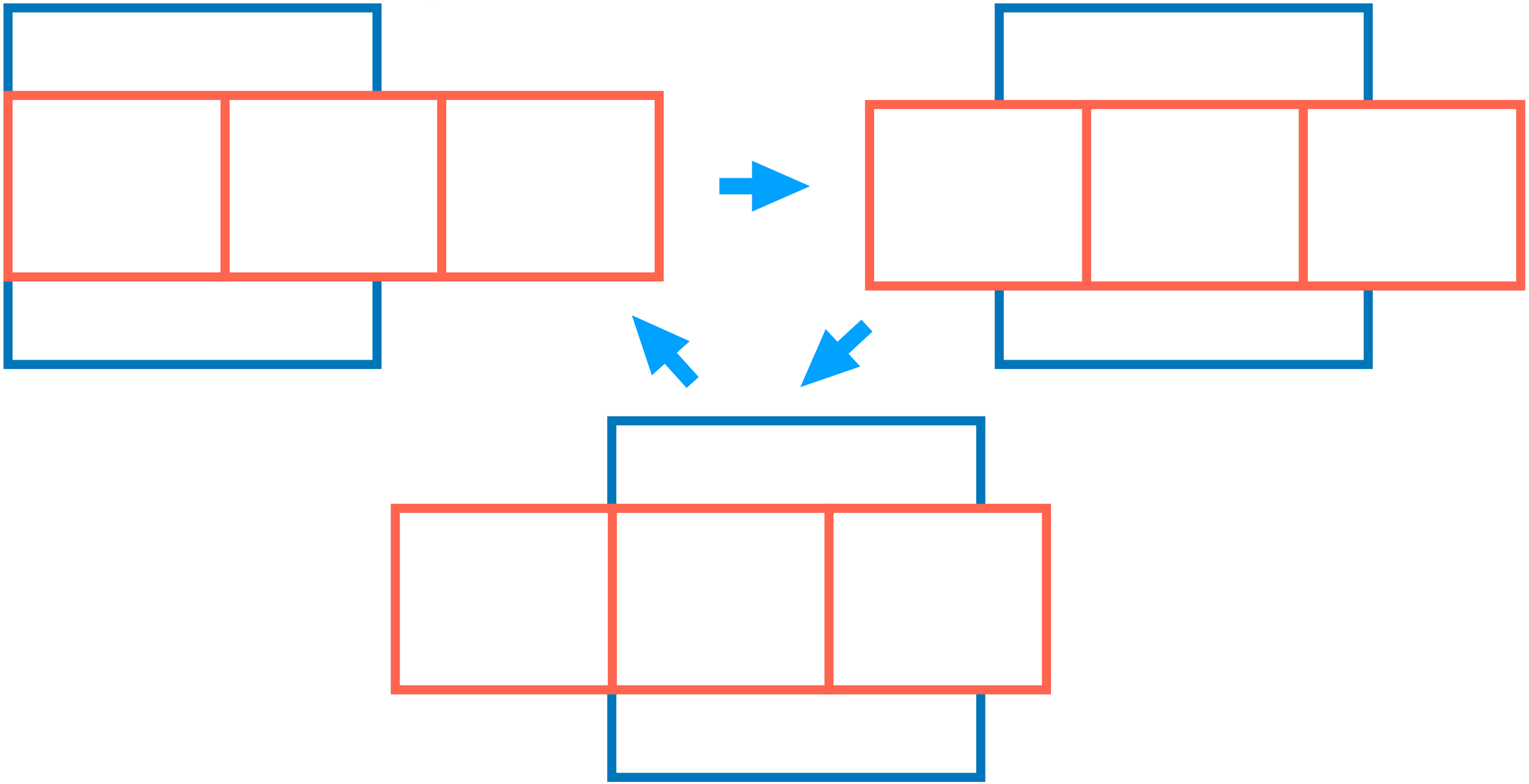
Image DOM (Document Object Model) Object



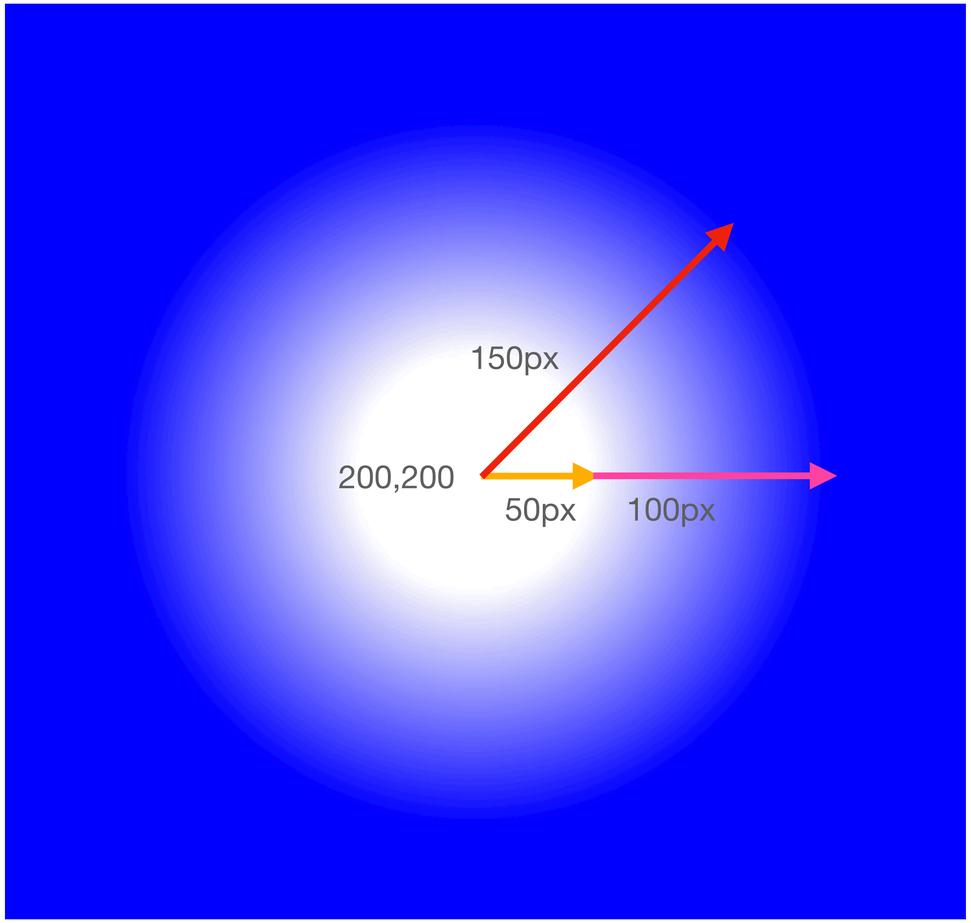
Measure text

Moving background

draw number of images

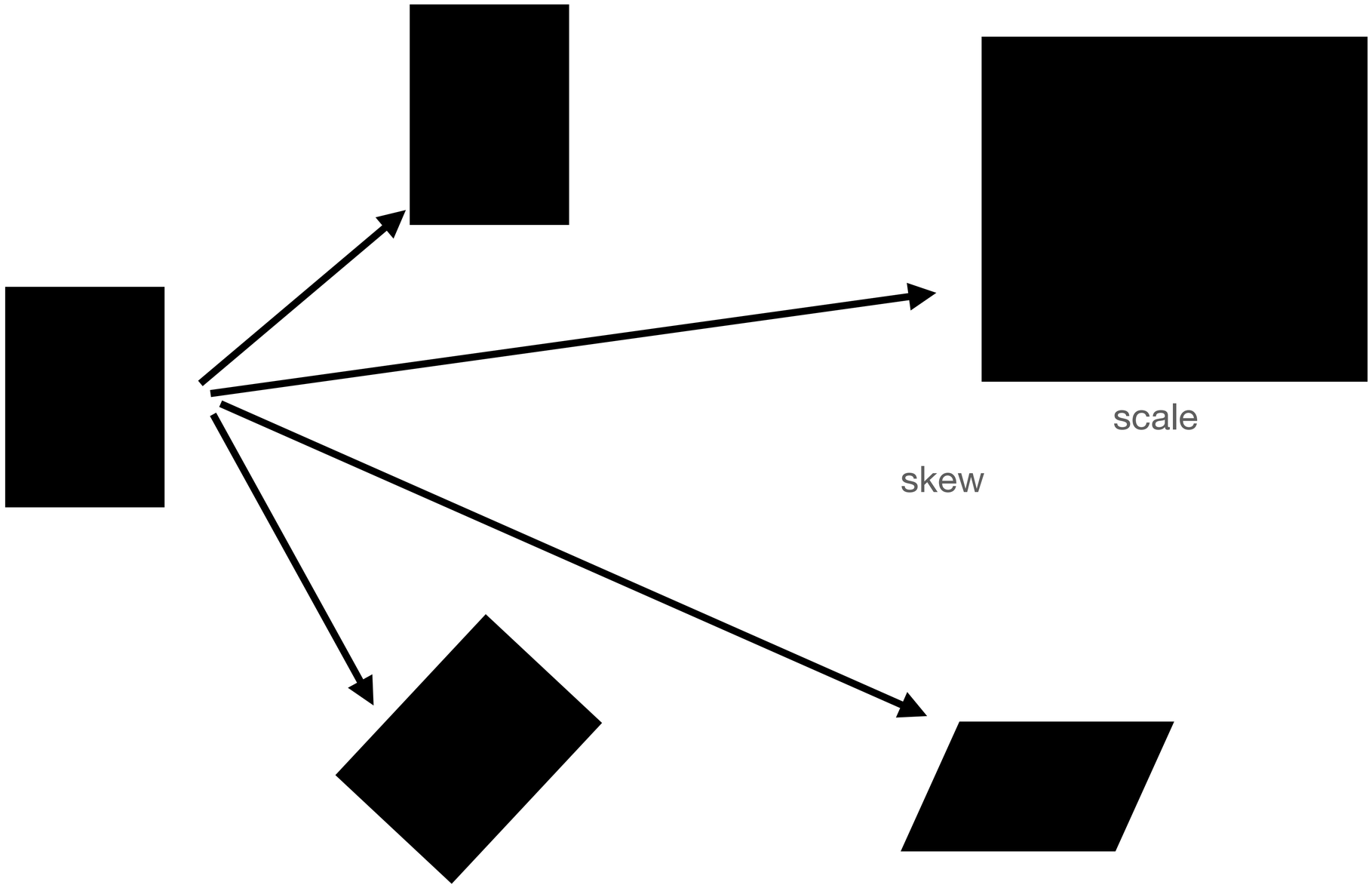


How radial gradient works?




```
g = ctx.createRadialGradient(200, 200, 50, 200, 200, 150);
g.addColorStop(0, 'white');
g.addColorStop(1, 'blue');
```

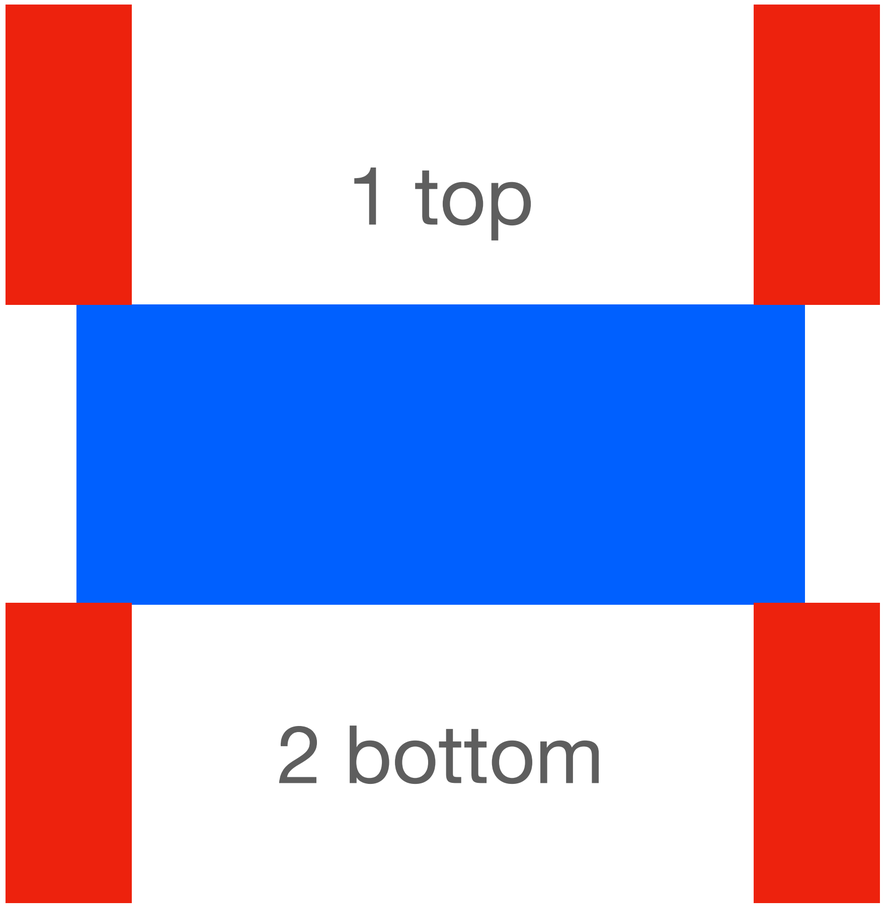
Transform



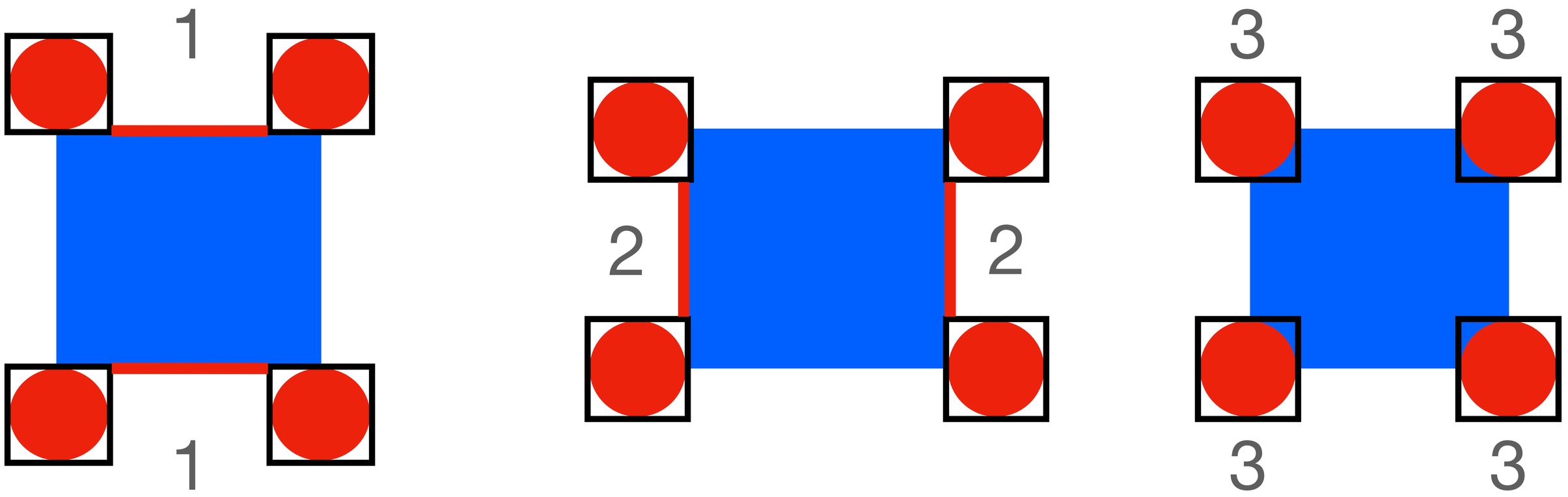
2D breakout

Collision detection
between circle and rectangle

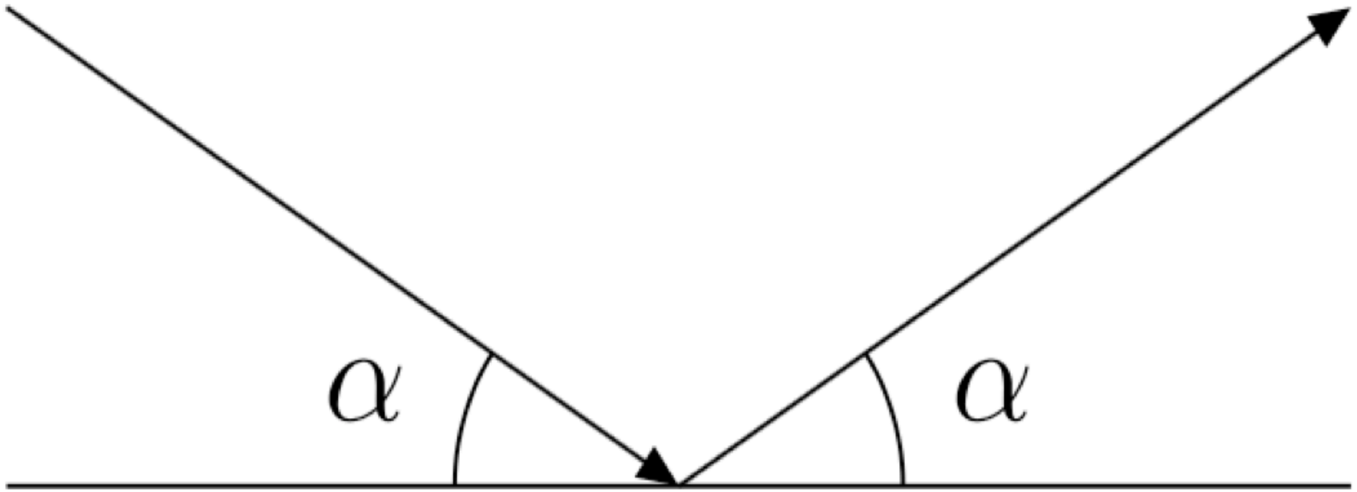
Collision detection between 2 rectangles



Collision detection between a circle and a rect



Relation between:
hit point -> angle

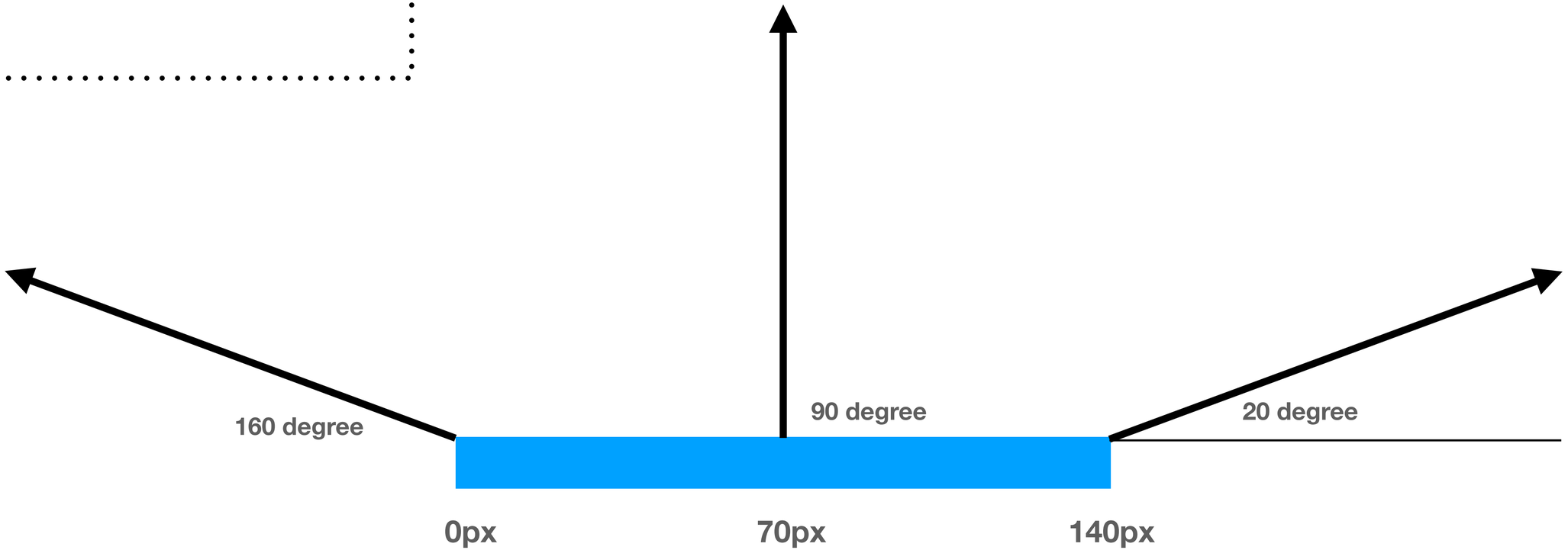

α
α

Relation between pixel and angle:
(160-20)/140=1

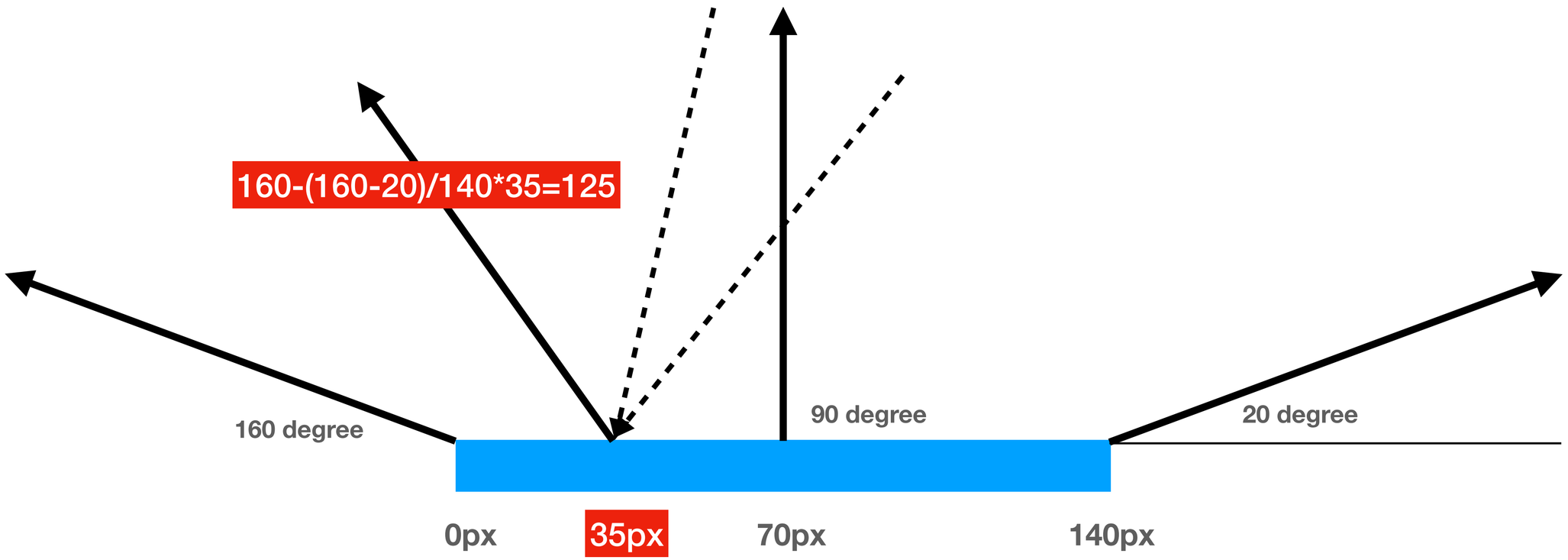
it means increasing 1px length,
decreasing 1 deg angle

example:
0px -> 160 deg
1px -> 159 deg
.....

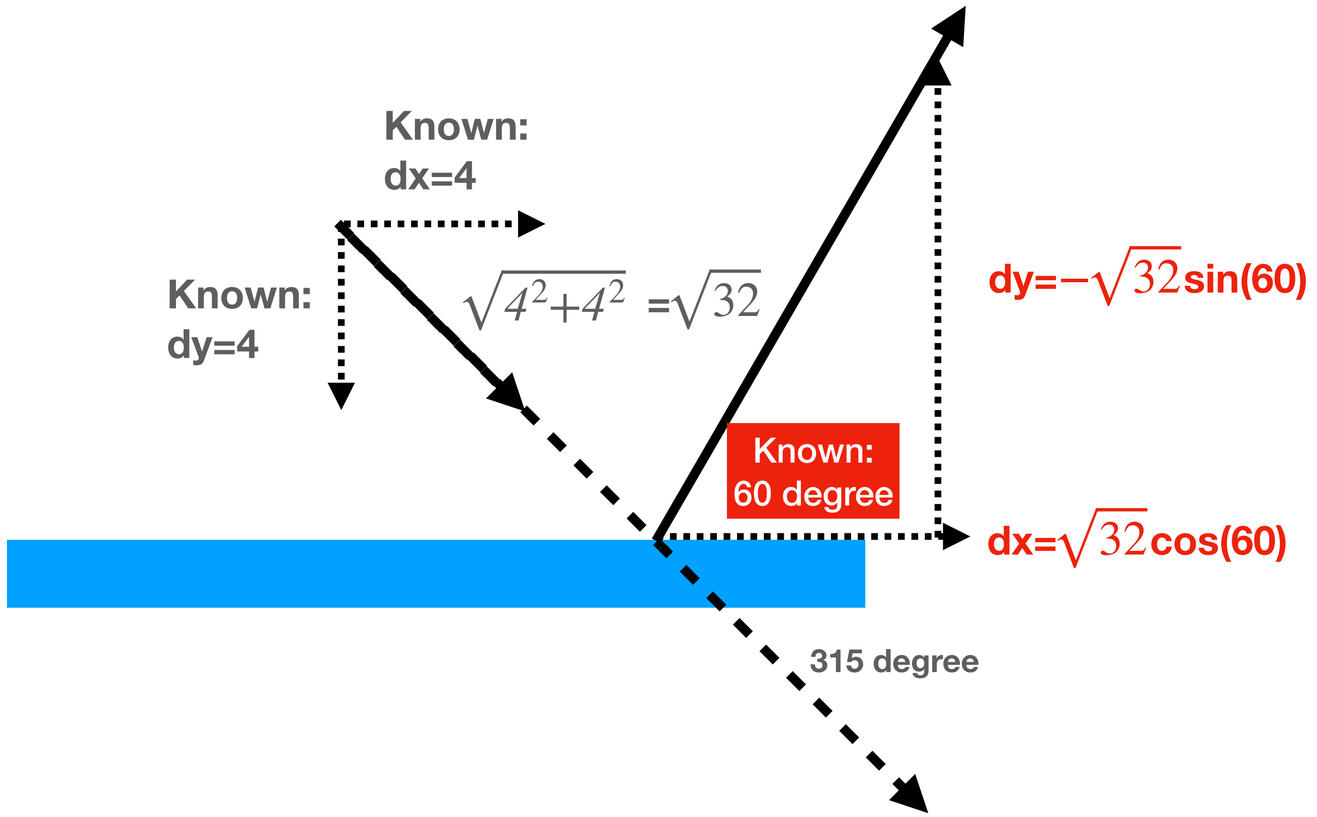
Formula:
angle=maxAngle-(maxAngle-minAngle)/paddleLength*paddleX



angle=maxAngle-(maxAngle-minAngle)/paddleLength*paddleX



Relation between:
dx, dy, Angle -> new dx, new dy


Known:
dx=4
Known:
dy=4
$\sqrt{4^2+4^2}$ =$\sqrt{32}$
dy=−$\sqrt{32}$sin(60)
Known:
60 degree
dx=$\sqrt{32}$cos(60)
315 degree