

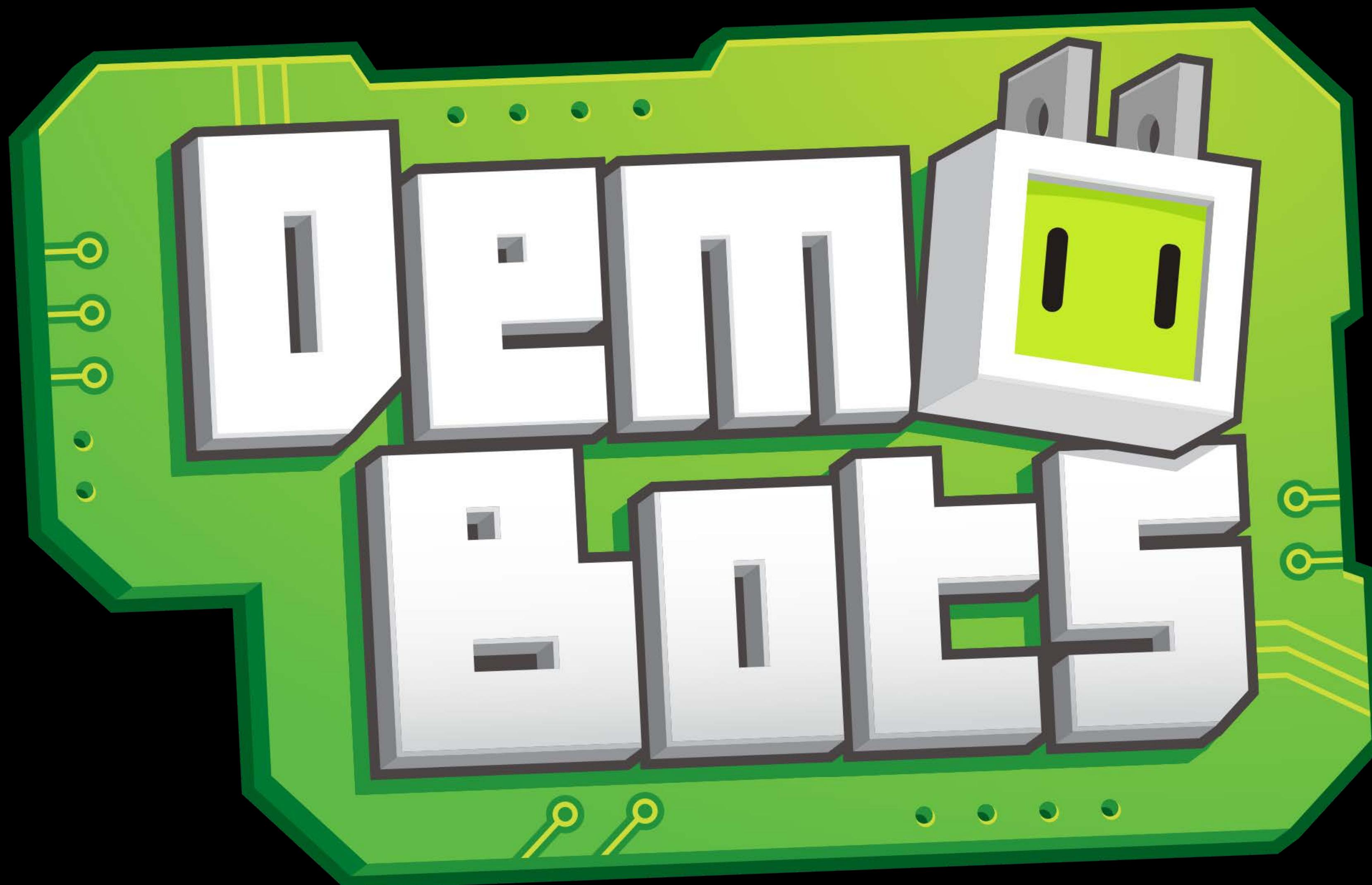
Deeper into GameplayKit with DemoBots

Session 609

Dave Addey Sample Code Engineer

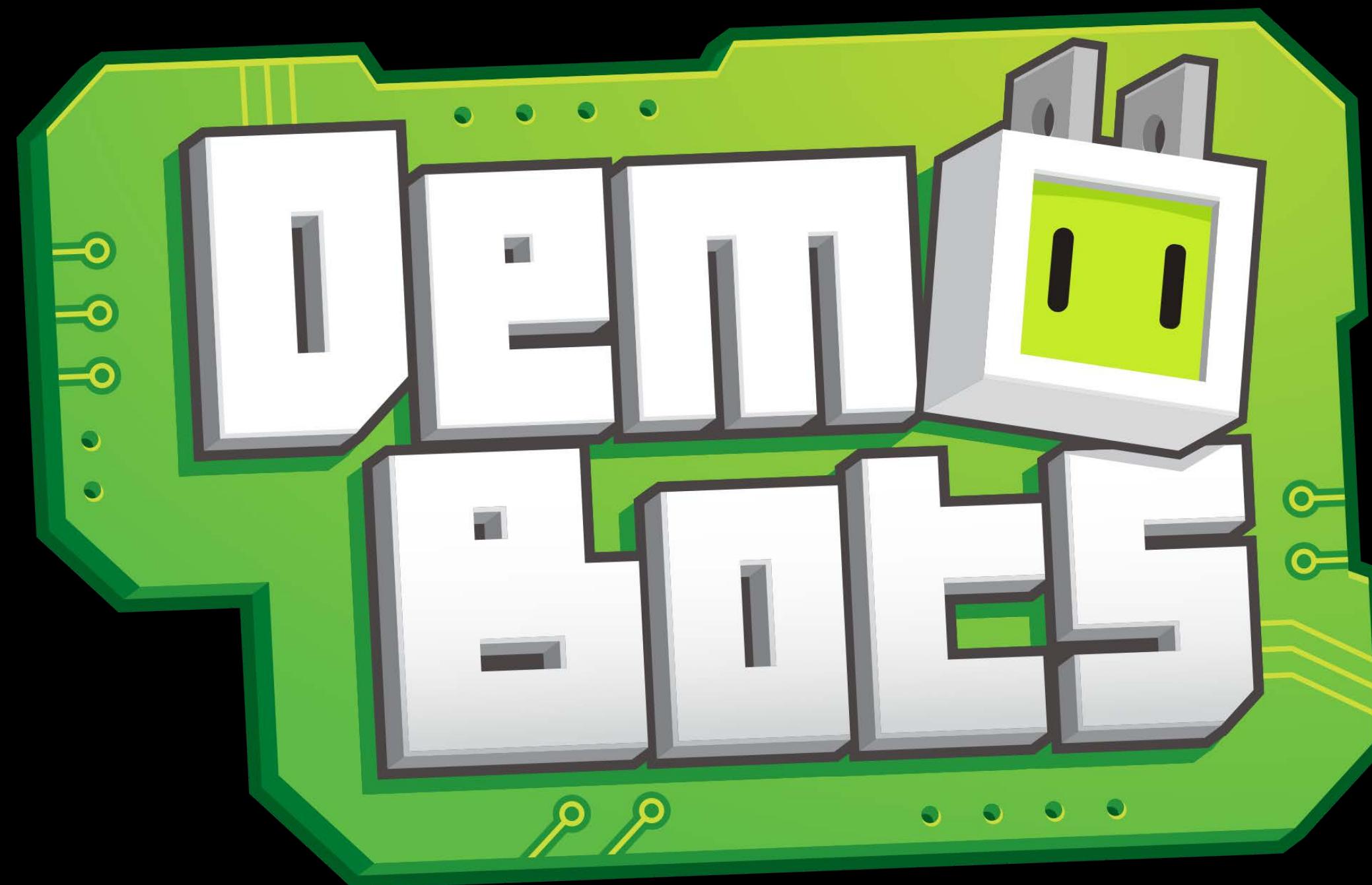
Dave Schaeffgen Sample Code Engineer

Michael DeWitt Sample Code Engineer



Download Sample Files

<http://developer.apple.com/spritekit>

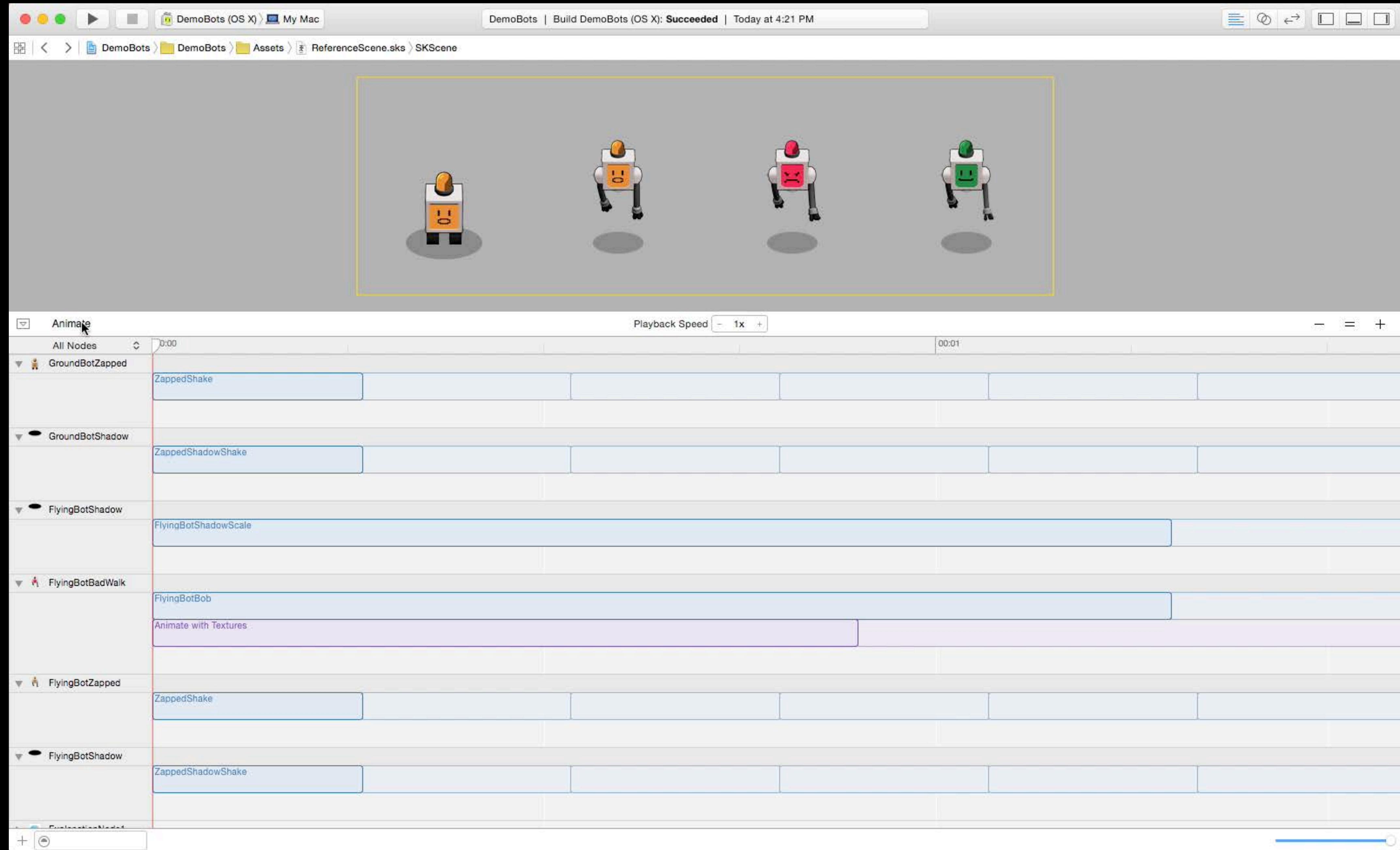


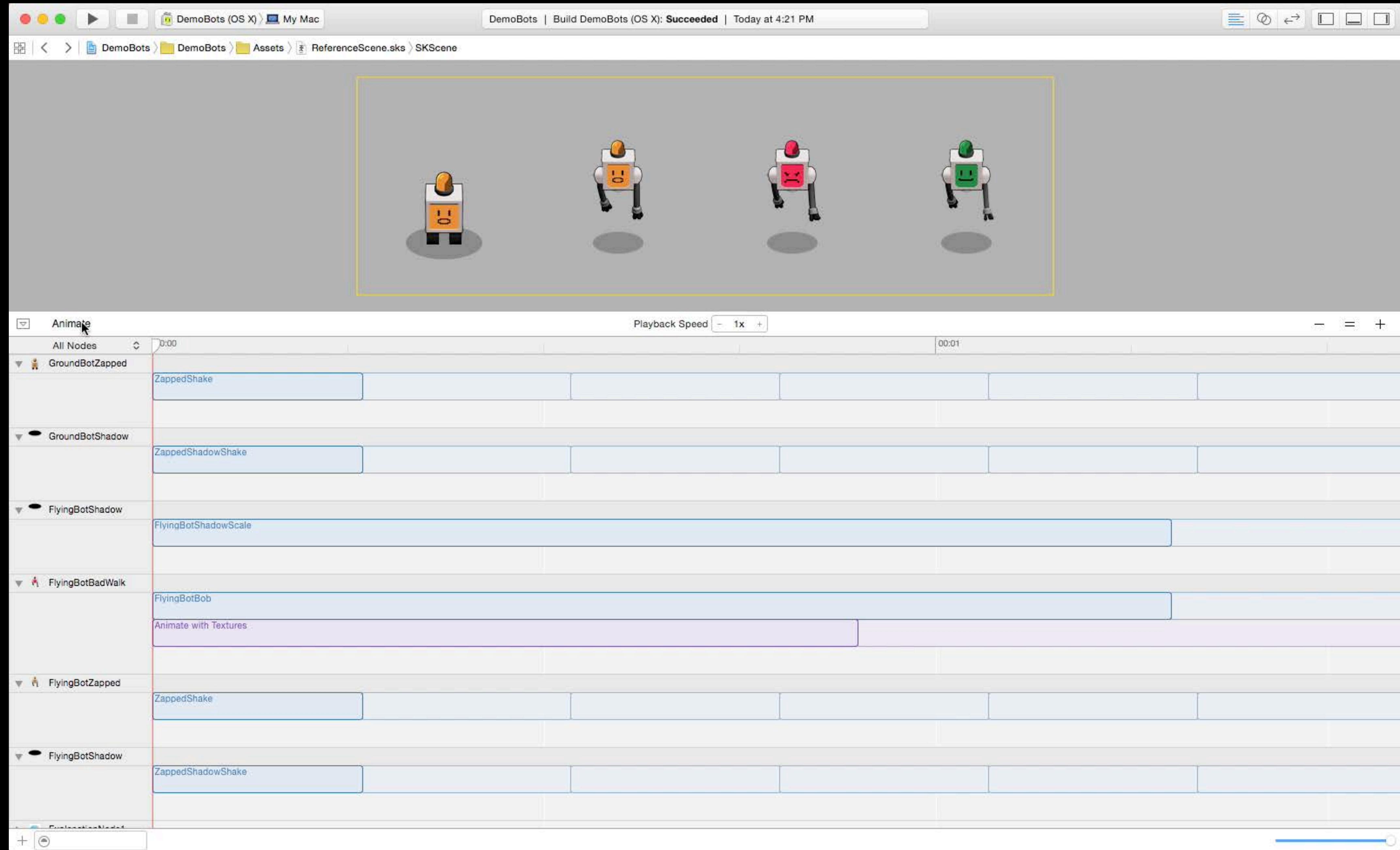
Recommended Sessions

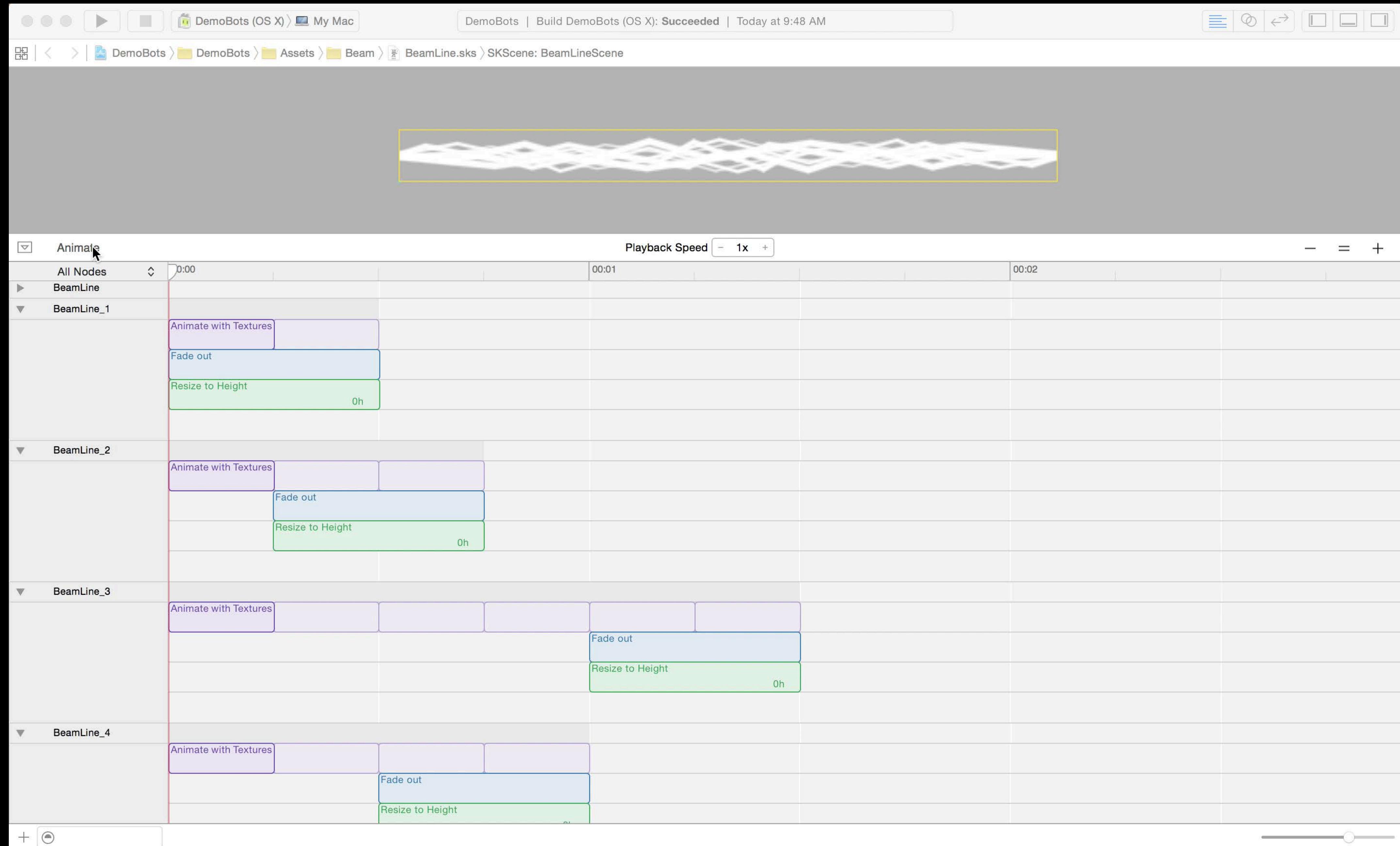
What's New in SpriteKit	Mission	Wednesday 10:00AM
Introducing GameplayKit	Mission	Thursday 11:00AM

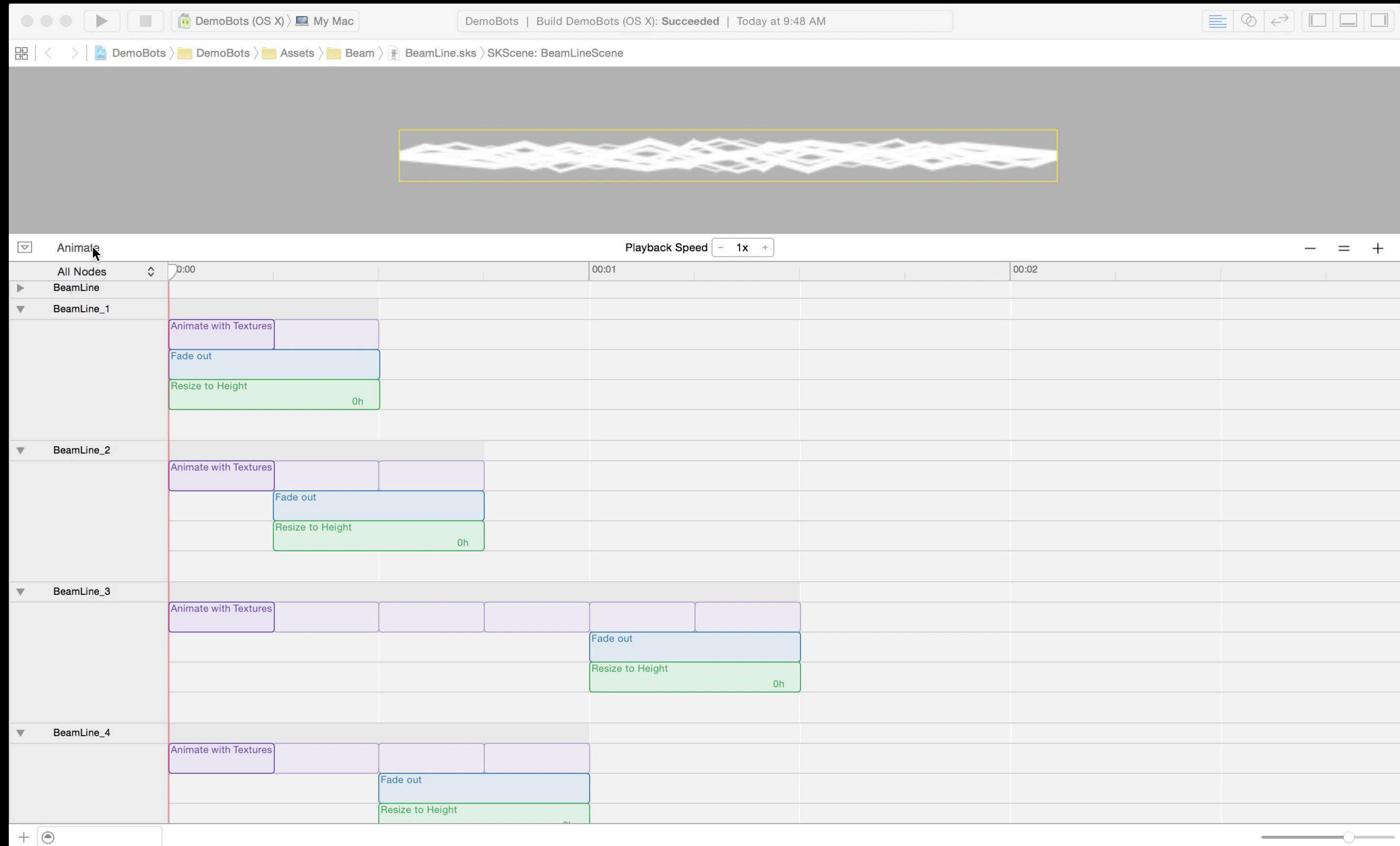
Demo

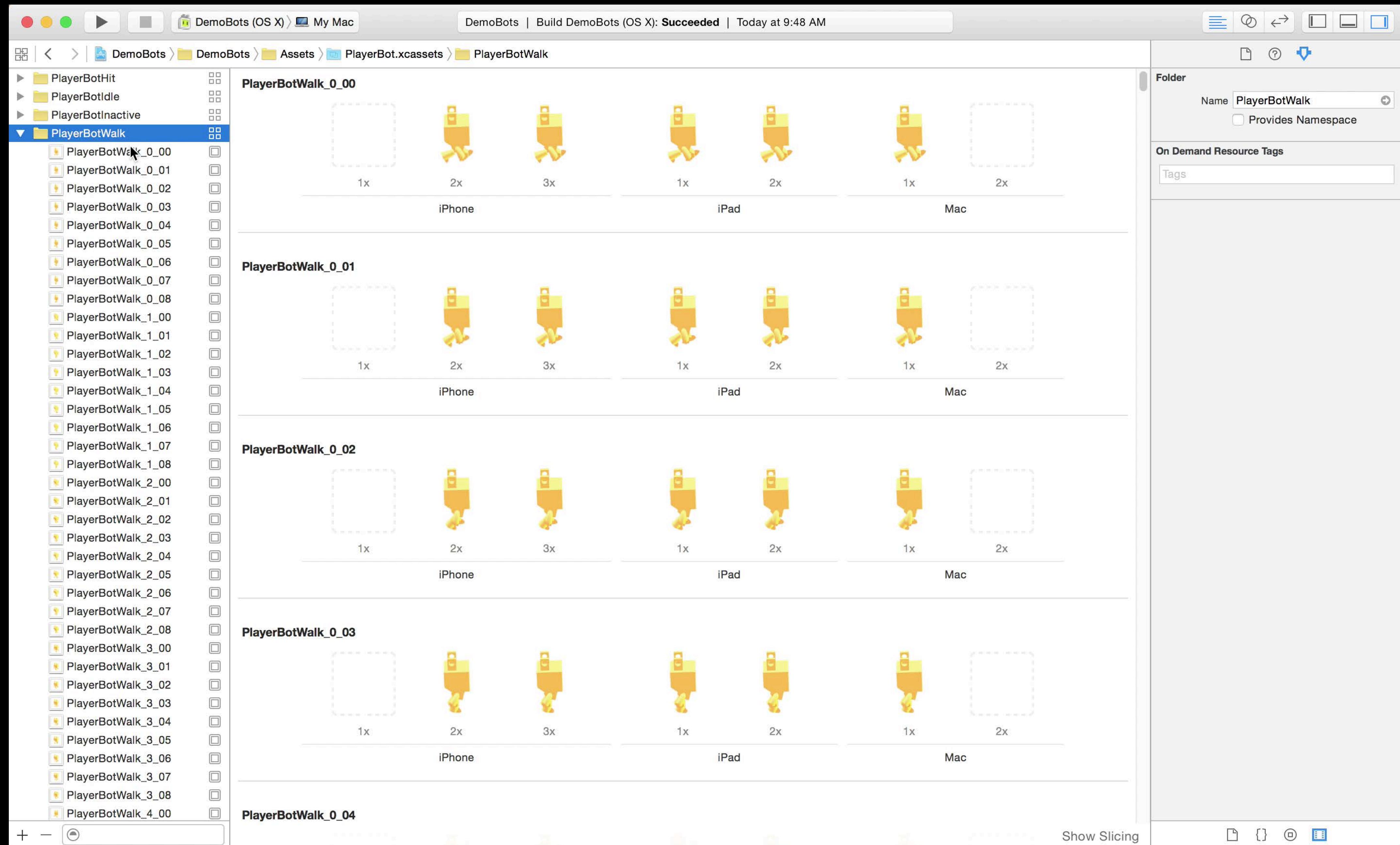
Tools and Technologies

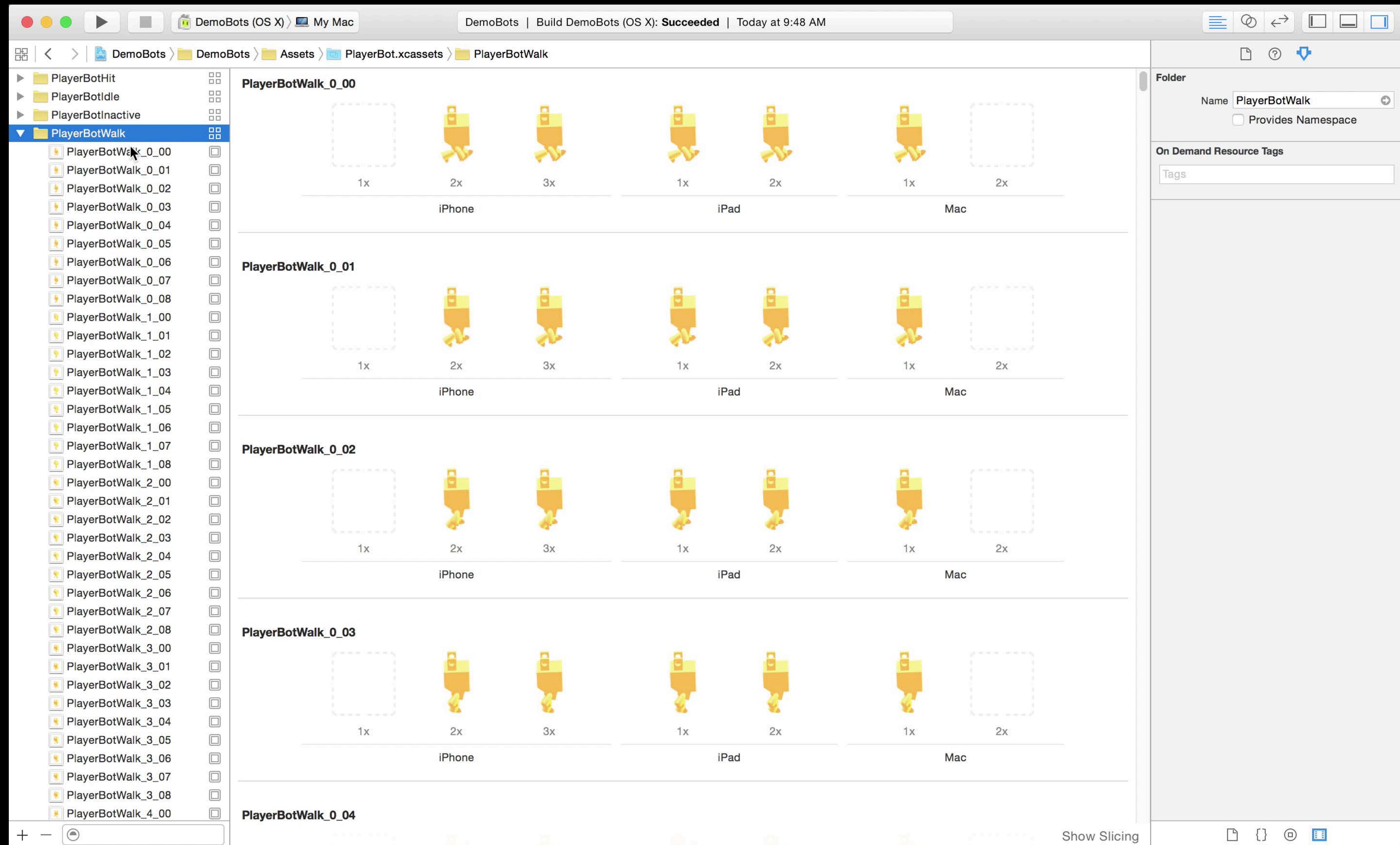


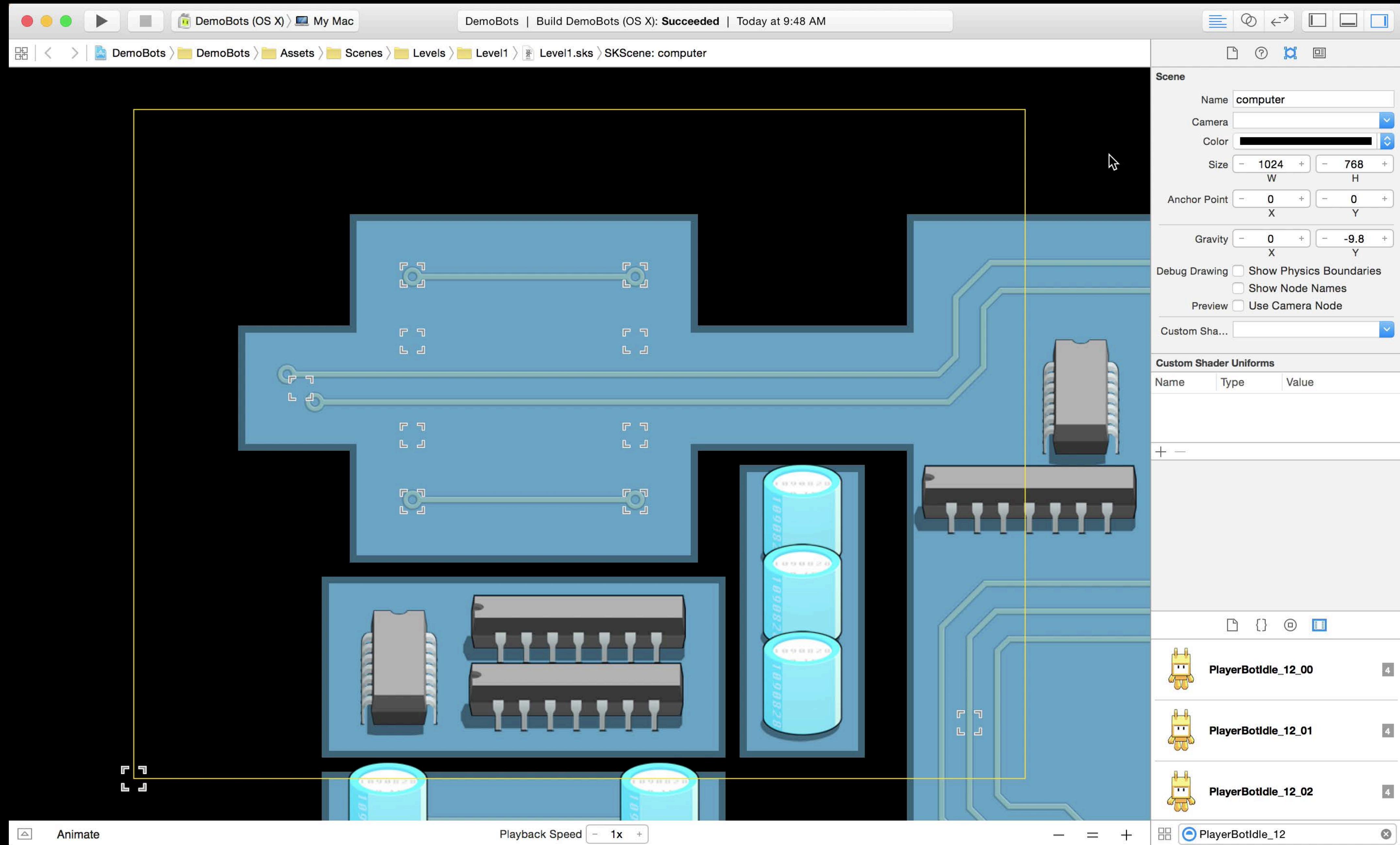


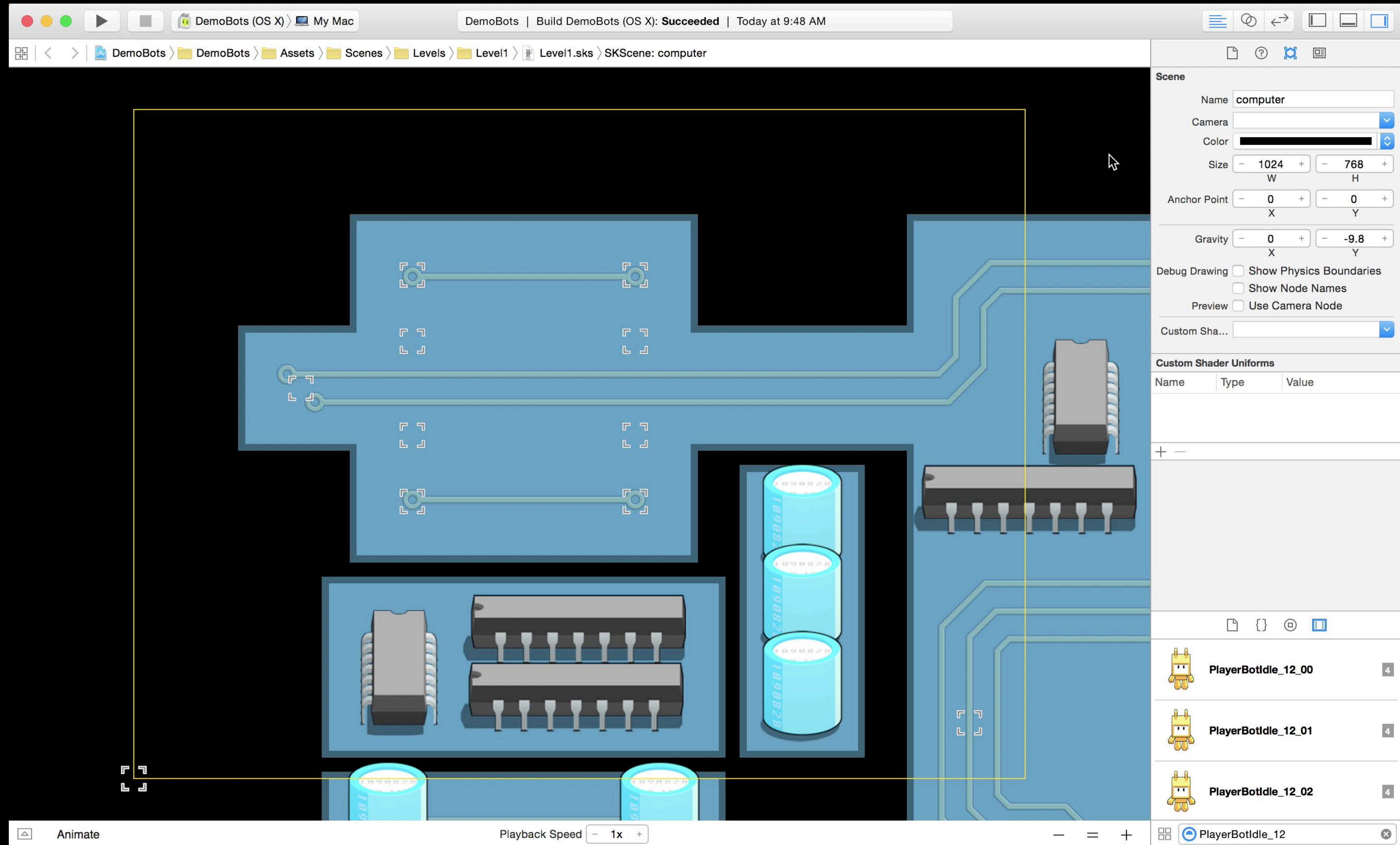


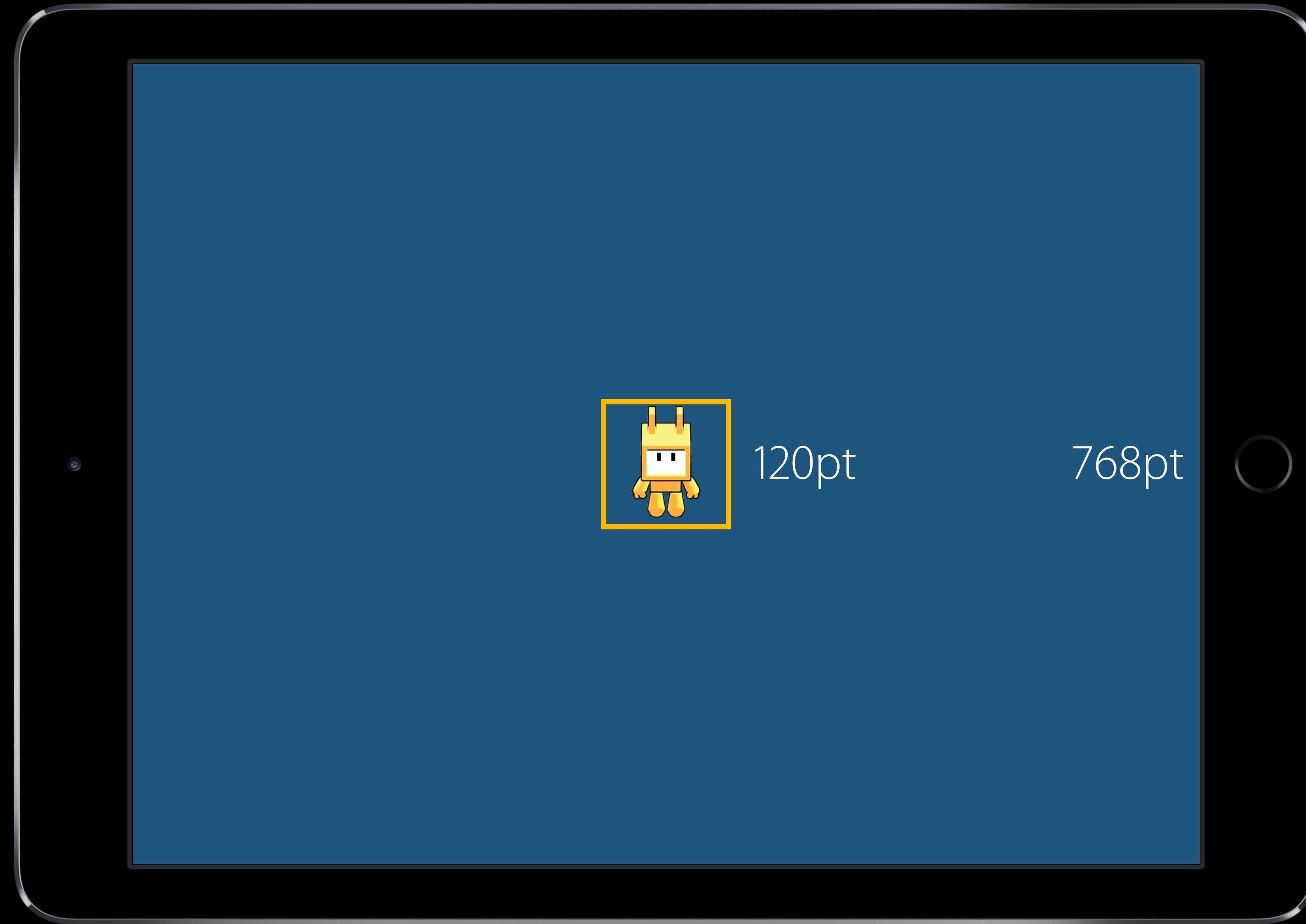


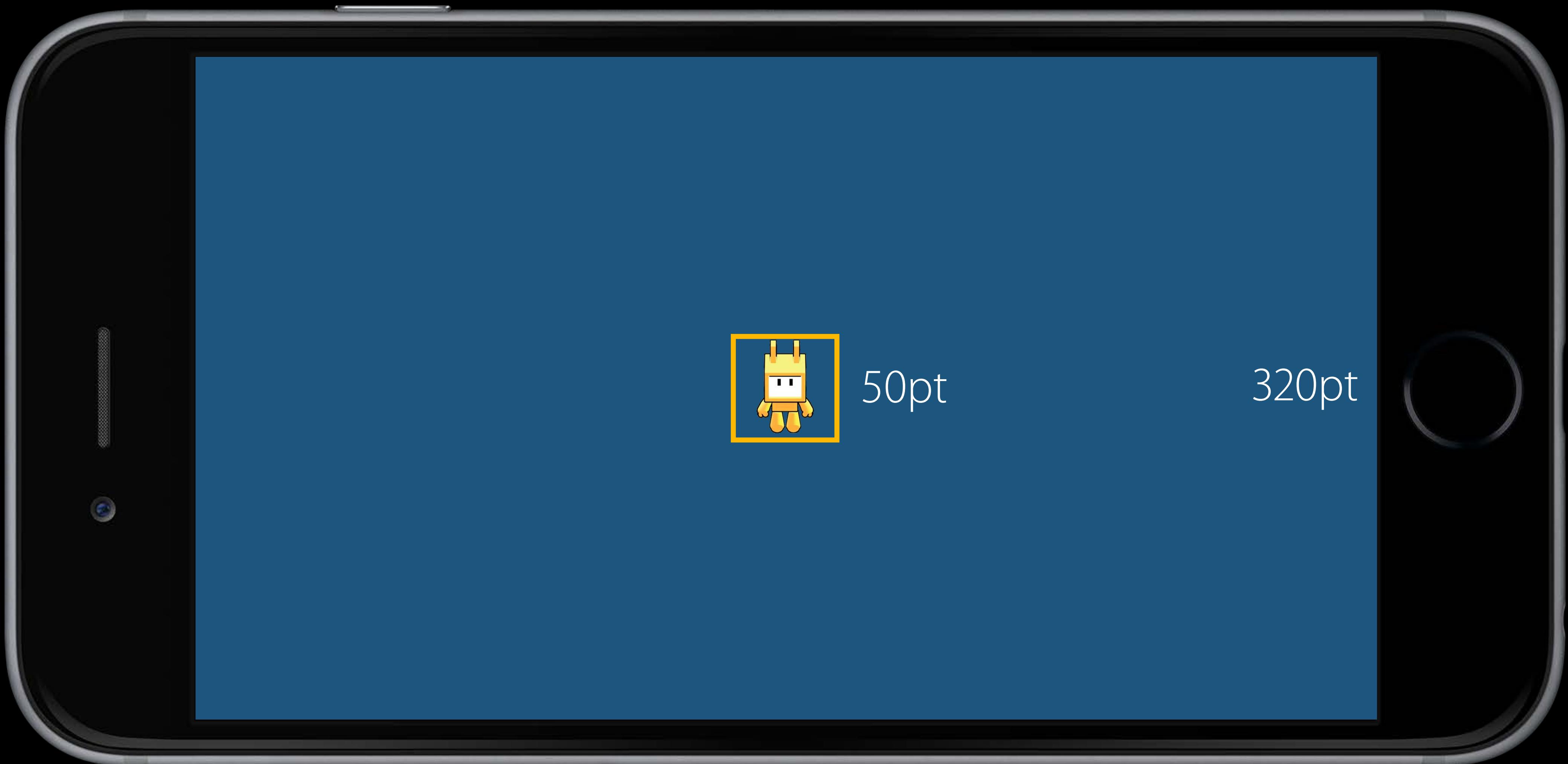












$$(320/768) \times 120 = 50\text{pt}$$

Optimizing Assets

Optimizing Assets

Device

iPad

iPhone 4s – 5s

iPhone 6

iPhone 6 Plus

Mac

Optimizing Assets

Device	Scene Height
--------	--------------

iPad	768pt
------	-------

iPhone 4s – 5s	320pt
----------------	-------

iPhone 6	326pt
----------	-------

iPhone 6 Plus	401pt
---------------	-------

Mac	1440pt
-----	--------

Optimizing Assets

Device	Scene Height	Player Height
iPad	768pt	120pt
iPhone 4s – 5s	320pt	50pt
iPhone 6	326pt	50pt
iPhone 6 Plus	401pt	62.65pt
Mac	1440pt	225pt

Optimizing Assets

Device	Scene Height	Player Height	1x
iPad	768pt	120pt	120px
iPhone 4s – 5s	320pt	50pt	-
iPhone 6	326pt	50pt	-
iPhone 6 Plus	401pt	62.65pt	-
Mac	1440pt	225pt	225px

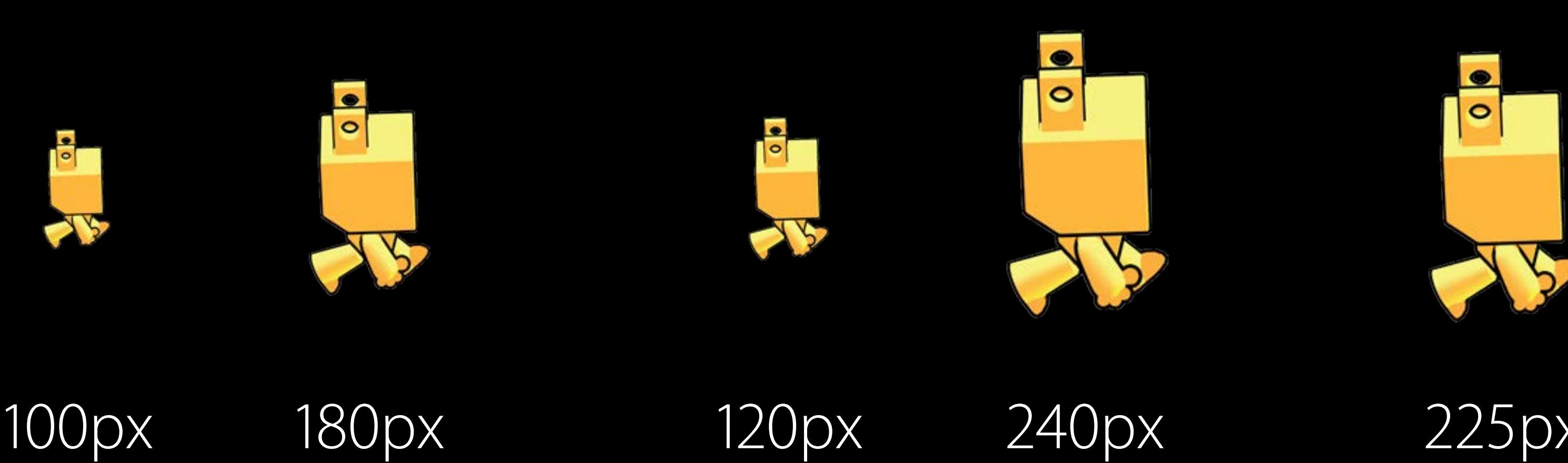
Optimizing Assets

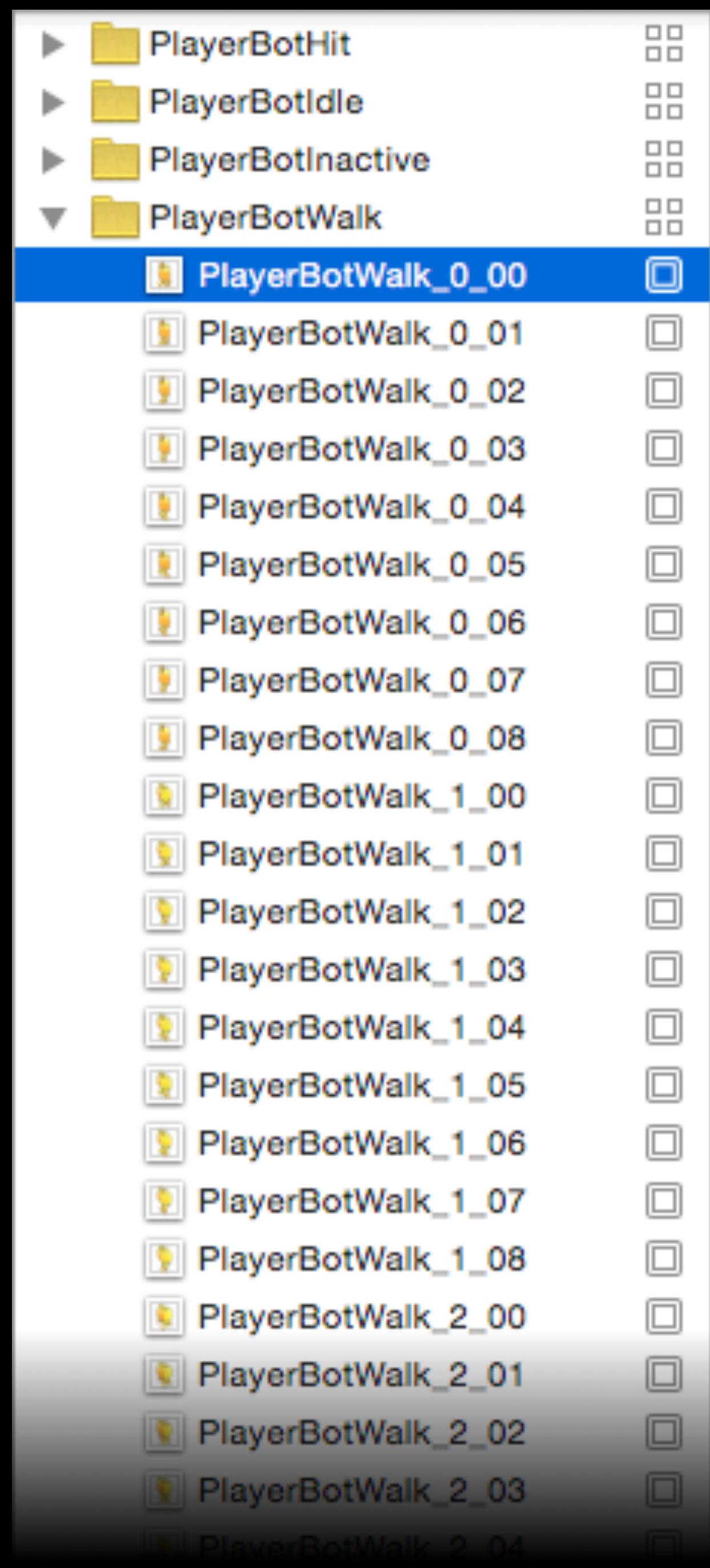
Device	Scene Height	Player Height	1x	2x
iPad	768pt	120pt	120px	240px
iPhone 4s – 5s	320pt	50pt	-	100px
iPhone 6	326pt	50pt	-	100px
iPhone 6 Plus	401pt	62.65pt	-	-
Mac	1440pt	225pt	225px	-

Optimizing Assets

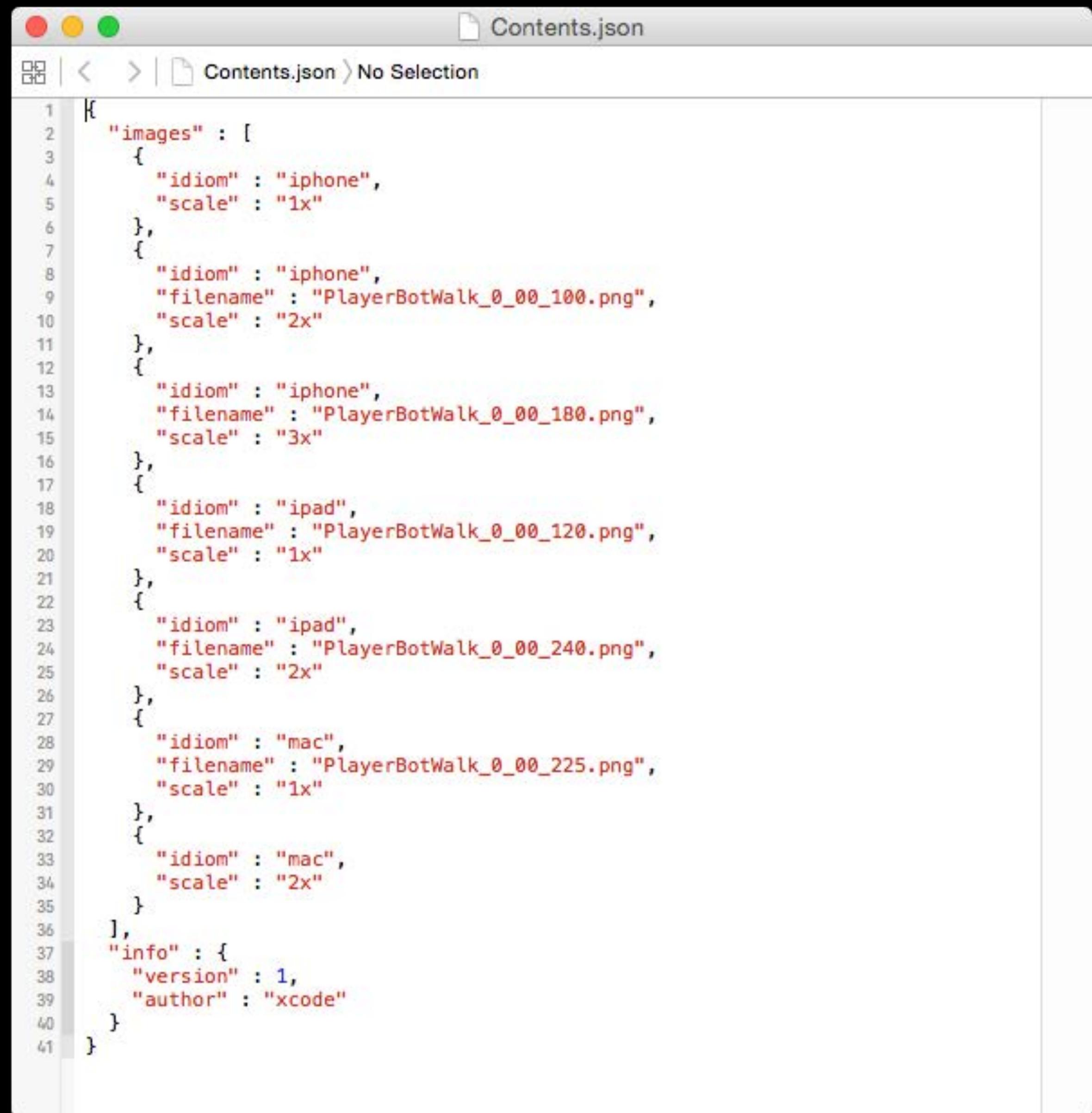
Device	Scene Height	Player Height	1x	2x	3x
iPad	768pt	120pt	120px	240px	-
iPhone 4s – 5s	320pt	50pt	-	100px	-
iPhone 6	326pt	50pt	-	100px	-
iPhone 6 Plus	401pt	62.65pt	-	-	180px
Mac	1440pt	225pt	225px	-	-

Optimizing Assets





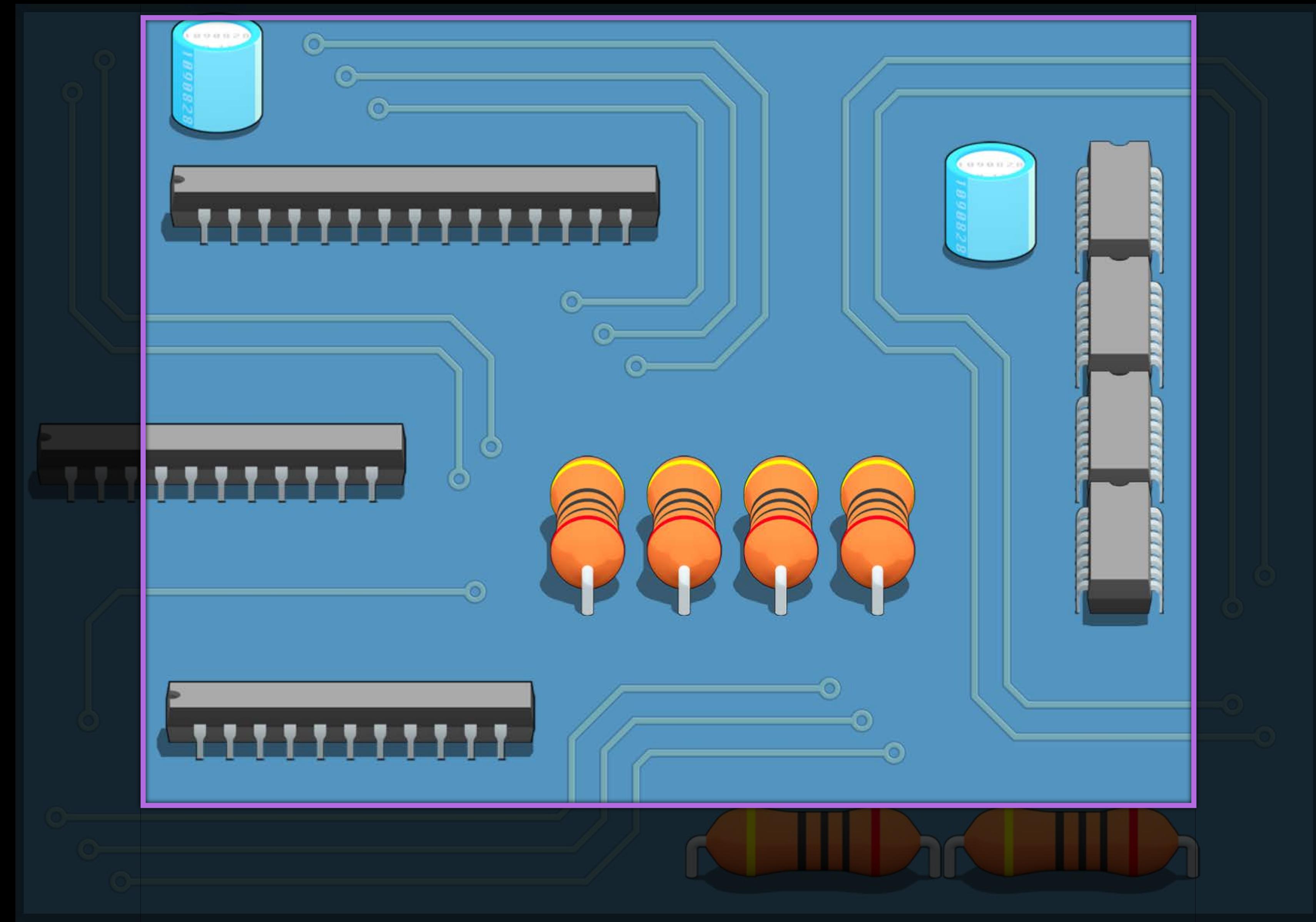
- PlayerBotWalk_13_01
- PlayerBotWalk_13_02
- PlayerBotWalk_13_03
- PlayerBotWalk_13_04
- PlayerBotWalk_13_05
- PlayerBotWalk_13_06
- PlayerBotWalk_13_07
- PlayerBotWalk_13_08
- PlayerBotWalk_14_00
- PlayerBotWalk_14_01
- PlayerBotWalk_14_02
- PlayerBotWalk_14_03
- PlayerBotWalk_14_04
- PlayerBotWalk_14_05
- PlayerBotWalk_14_06
- PlayerBotWalk_14_07
- PlayerBotWalk_14_08
- PlayerBotWalk_15_00
- PlayerBotWalk_15_01
- PlayerBotWalk_15_02
- PlayerBotWalk_15_03
- PlayerBotWalk_15_04
- PlayerBotWalk_15_05
- PlayerBotWalk_15_06
- PlayerBotWalk_15_07
- PlayerBotWalk_15_08



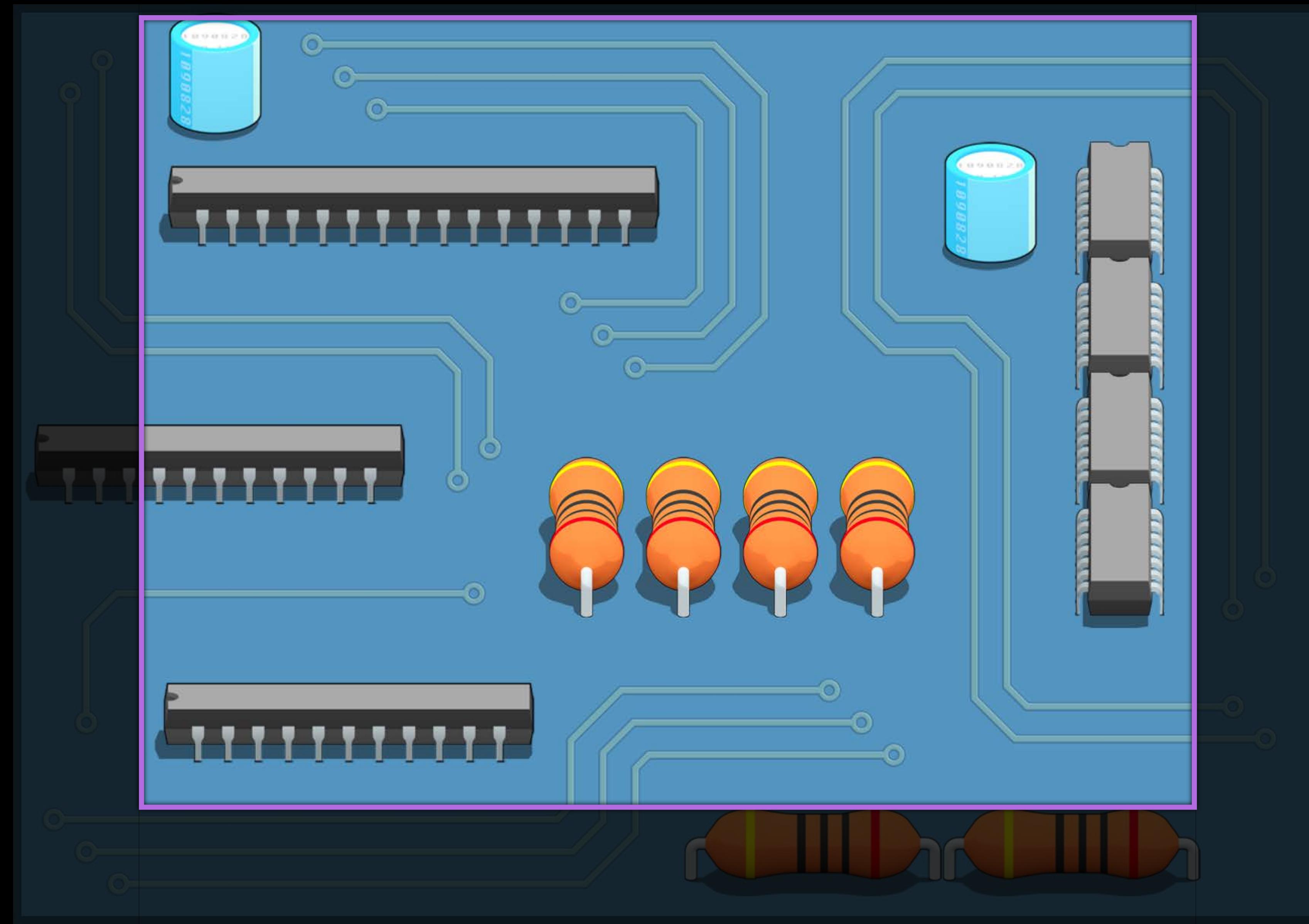
The screenshot shows a Mac OS X TextEdit window with a white background and a light gray header bar. The title bar reads "Contents.json". Below the title bar is a toolbar with standard Mac OS X icons for file operations. The main area of the window contains a JSON document with line numbers on the left side.

```
1  {
2      "images" : [
3          {
4              "idiom" : "iphone",
5              "scale" : "1x"
6          },
7          {
8              "idiom" : "iphone",
9              "filename" : "PlayerBotWalk_0_00_100.png",
10             "scale" : "2x"
11         },
12         {
13             "idiom" : "iphone",
14             "filename" : "PlayerBotWalk_0_00_180.png",
15             "scale" : "3x"
16         },
17         {
18             "idiom" : "ipad",
19             "filename" : "PlayerBotWalk_0_00_120.png",
20             "scale" : "1x"
21         },
22         {
23             "idiom" : "ipad",
24             "filename" : "PlayerBotWalk_0_00_240.png",
25             "scale" : "2x"
26         },
27         {
28             "idiom" : "mac",
29             "filename" : "PlayerBotWalk_0_00_225.png",
30             "scale" : "1x"
31         },
32         {
33             "idiom" : "mac",
34             "scale" : "2x"
35         }
36     ],
37     "info" : {
38         "version" : 1,
39         "author" : "xcode"
40     }
41 }
```

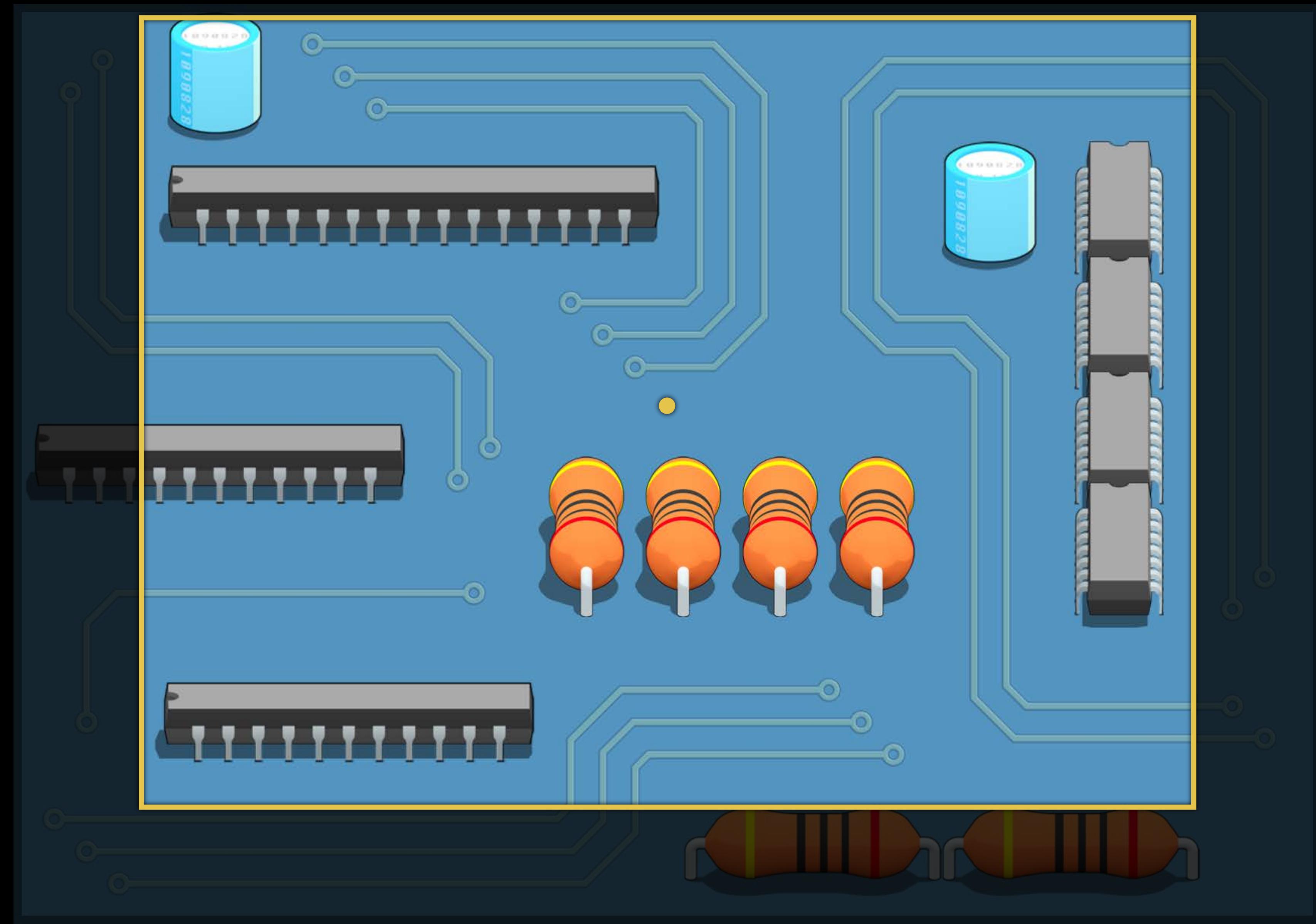
Before SKCameraNode



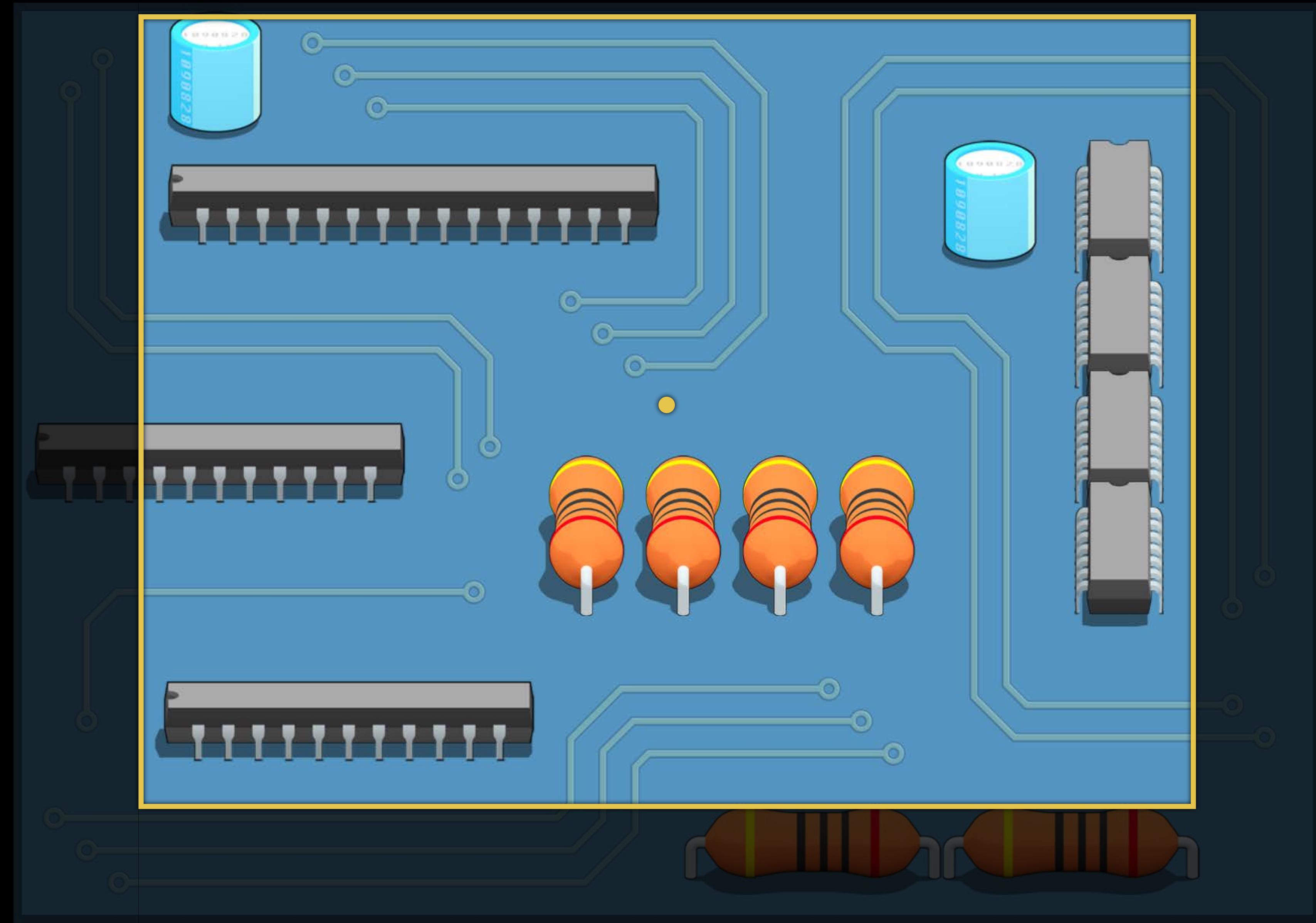
Before SKCameraNode



With SKCameraNode



With SKCameraNode



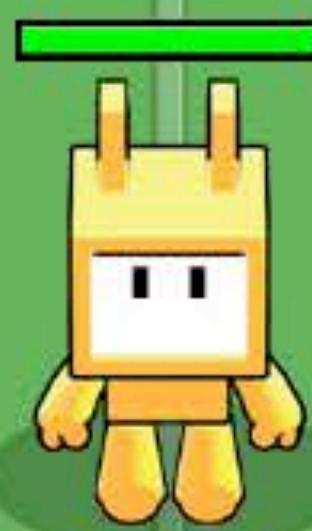
3:59



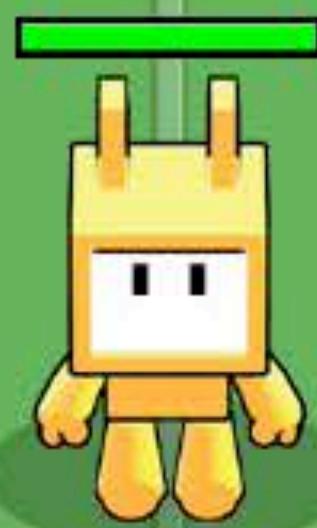
3:59

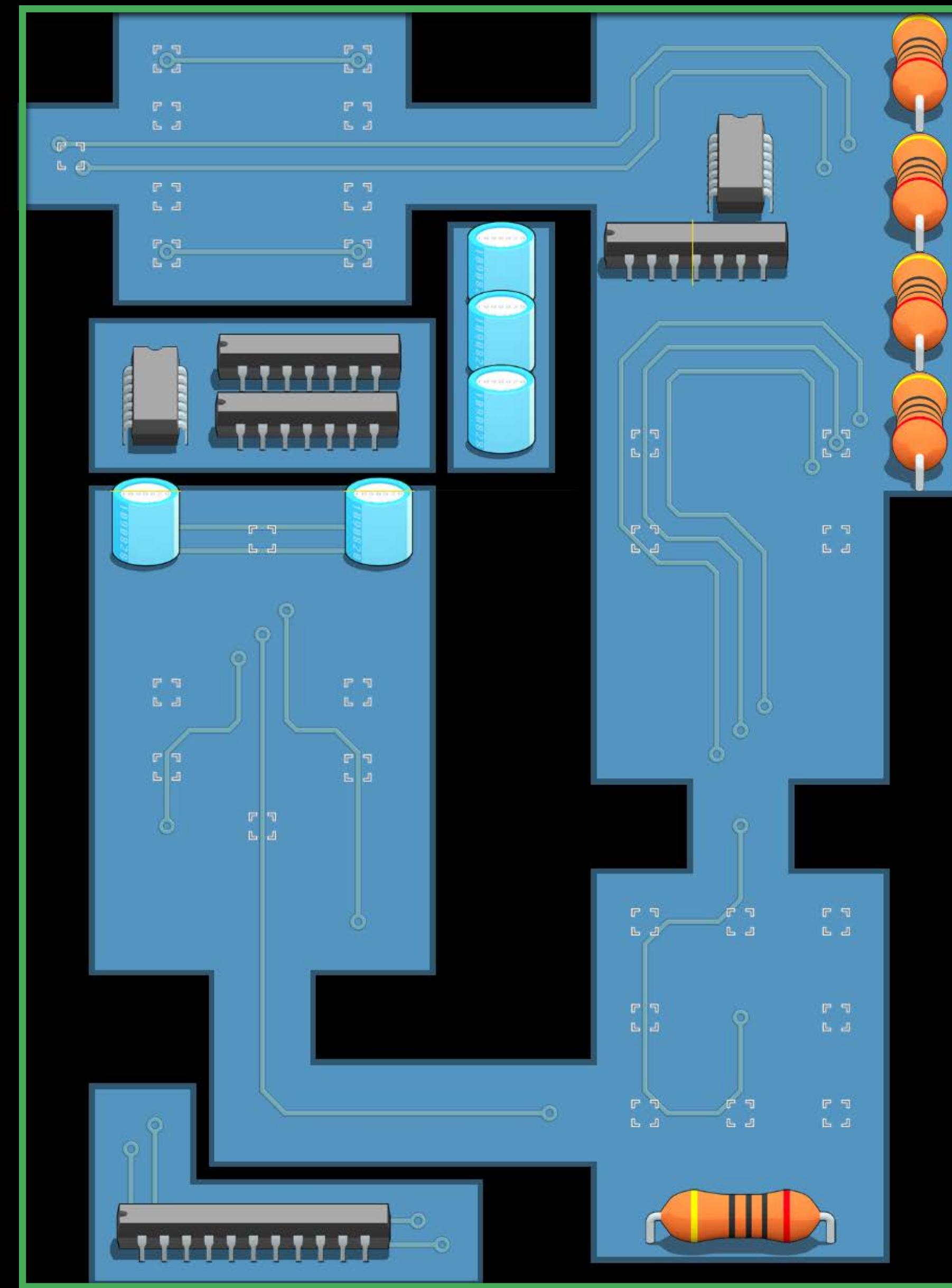


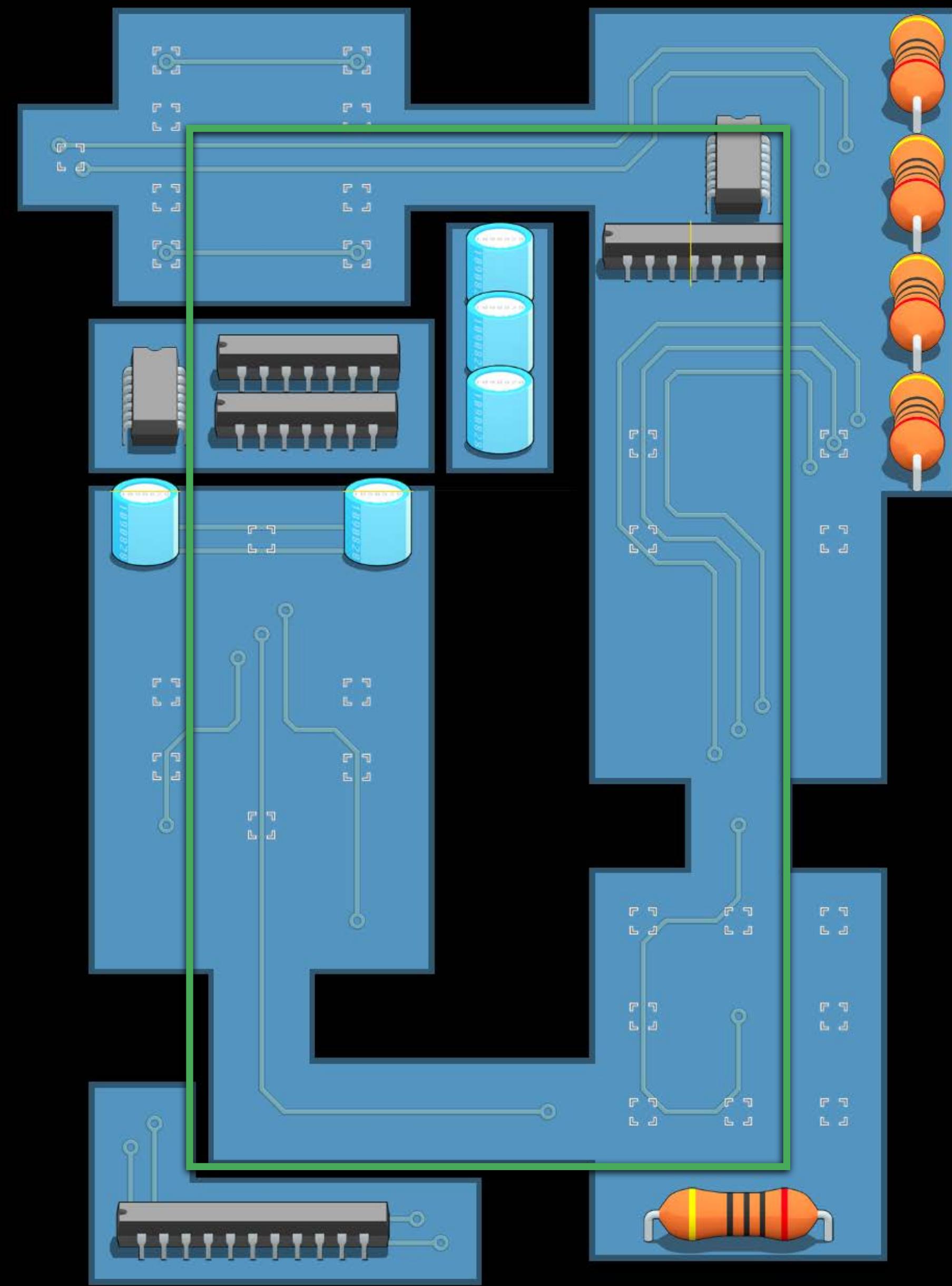
3:59

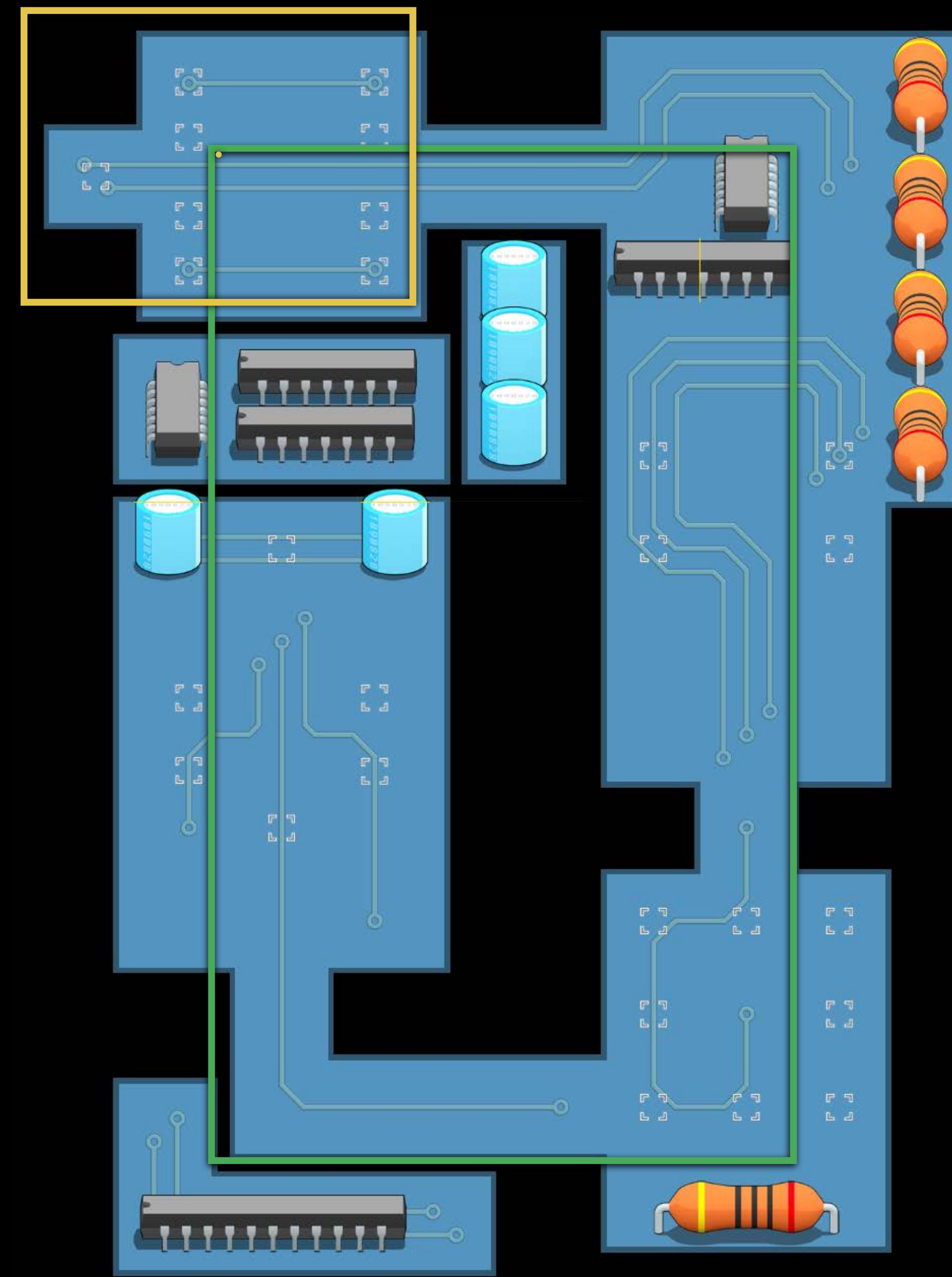


3:59

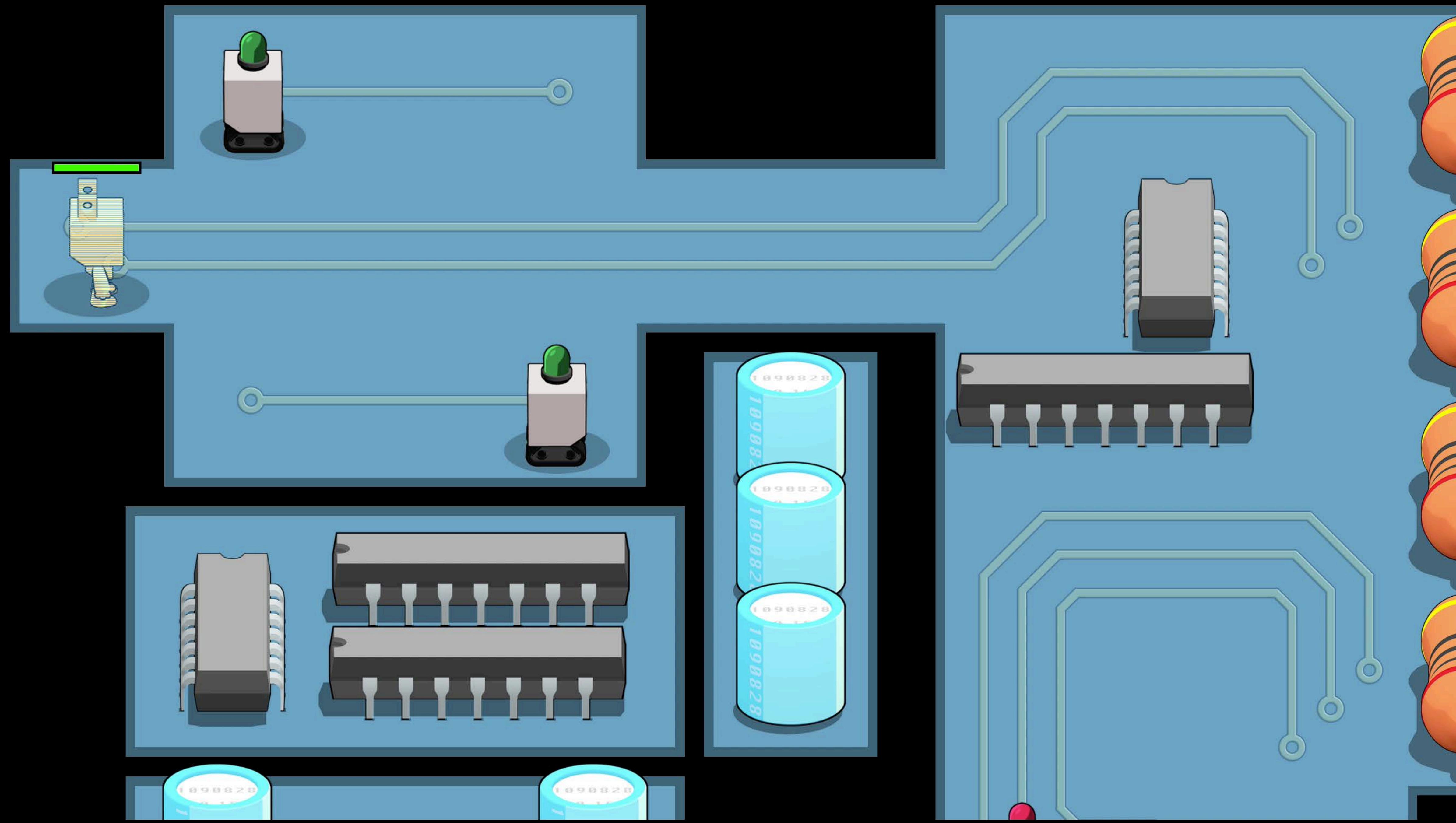




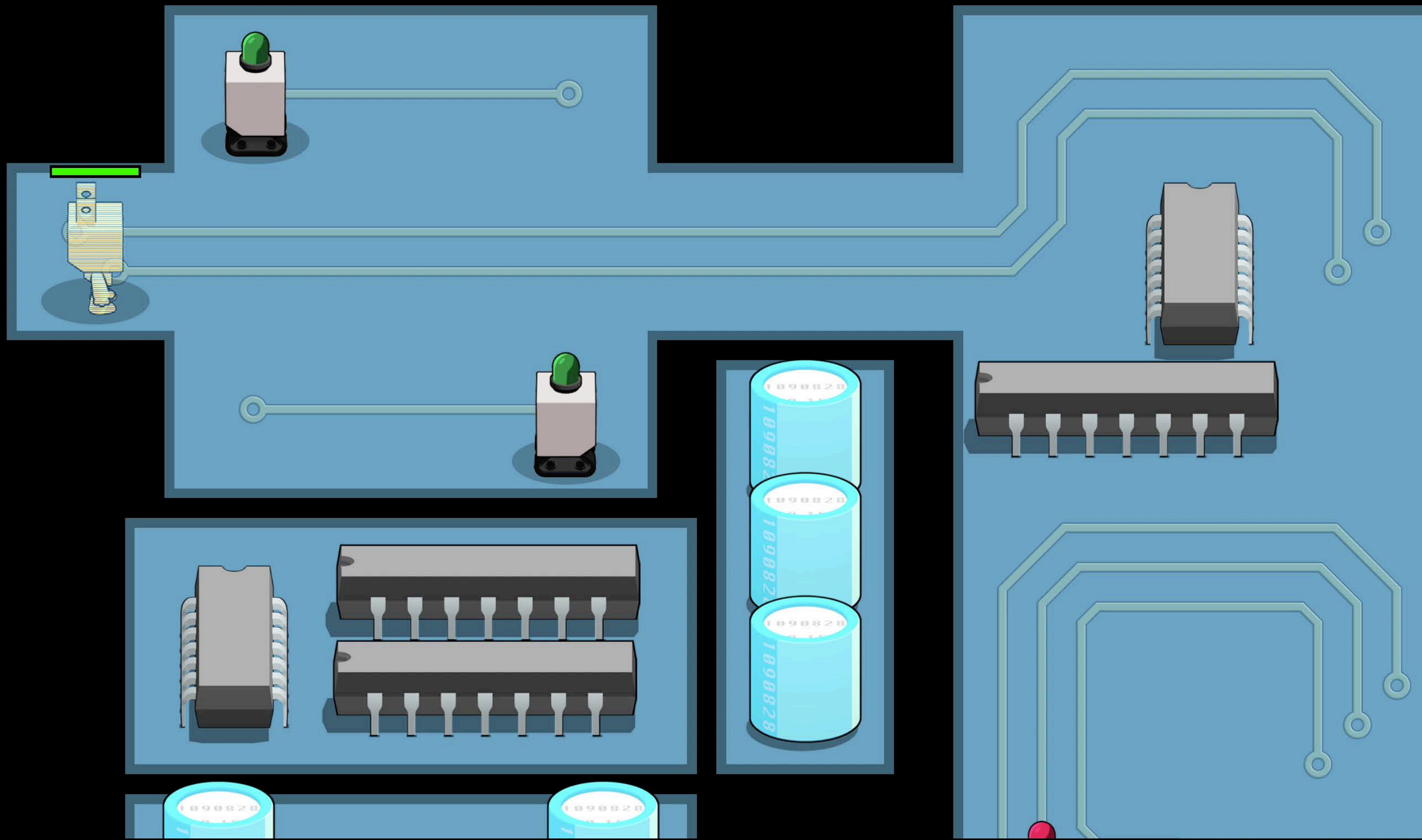




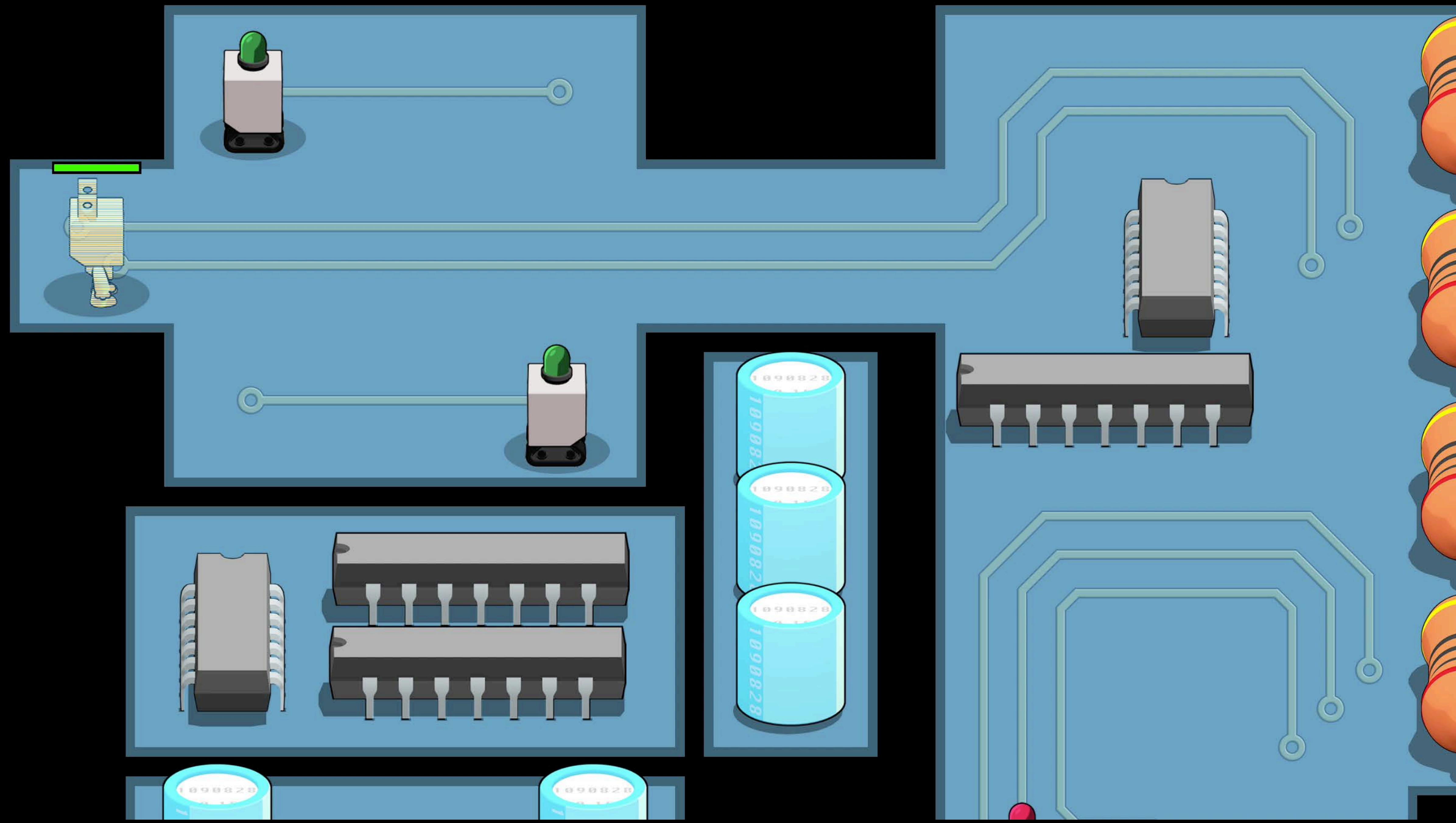
2:59



2:59



2:59



GKStateMachine

PlayerBot state



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging

GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging

GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

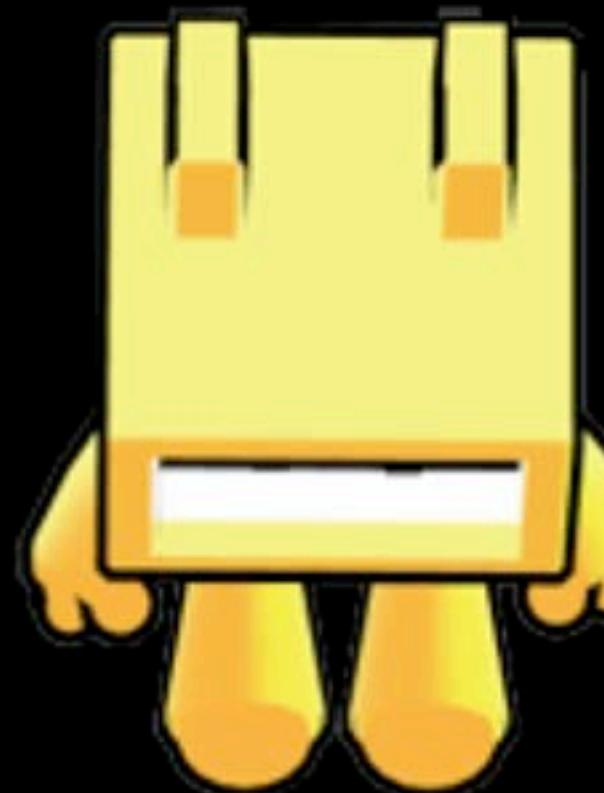
PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

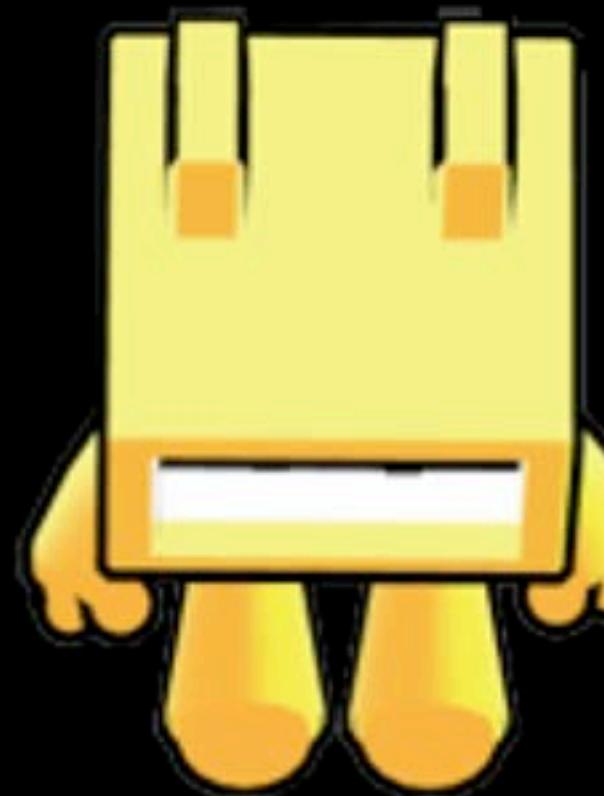
PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

Player Controlled

Hit

Recharging



GKStateMachine

PlayerBot state

Appear

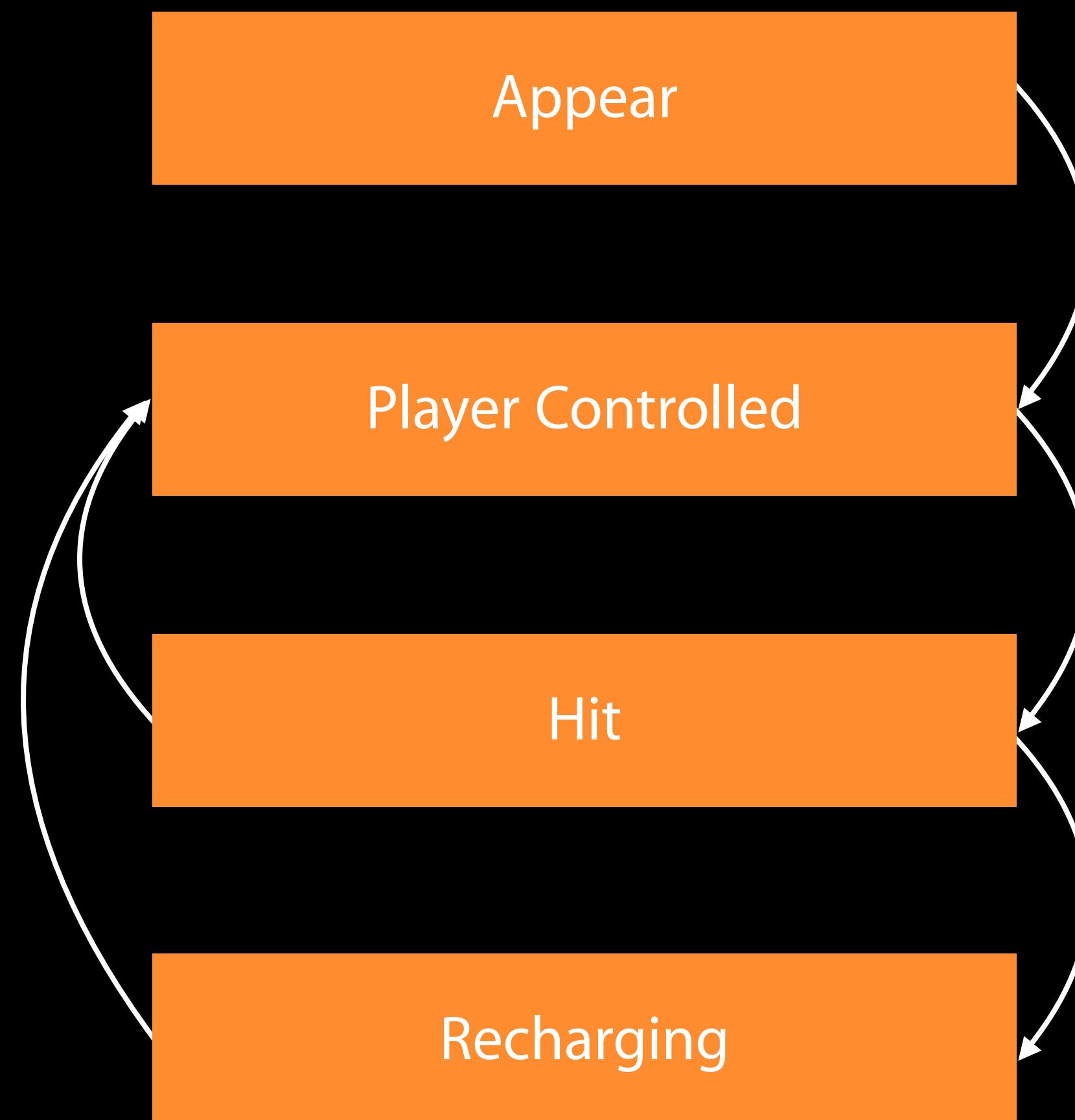
Player Controlled

Hit

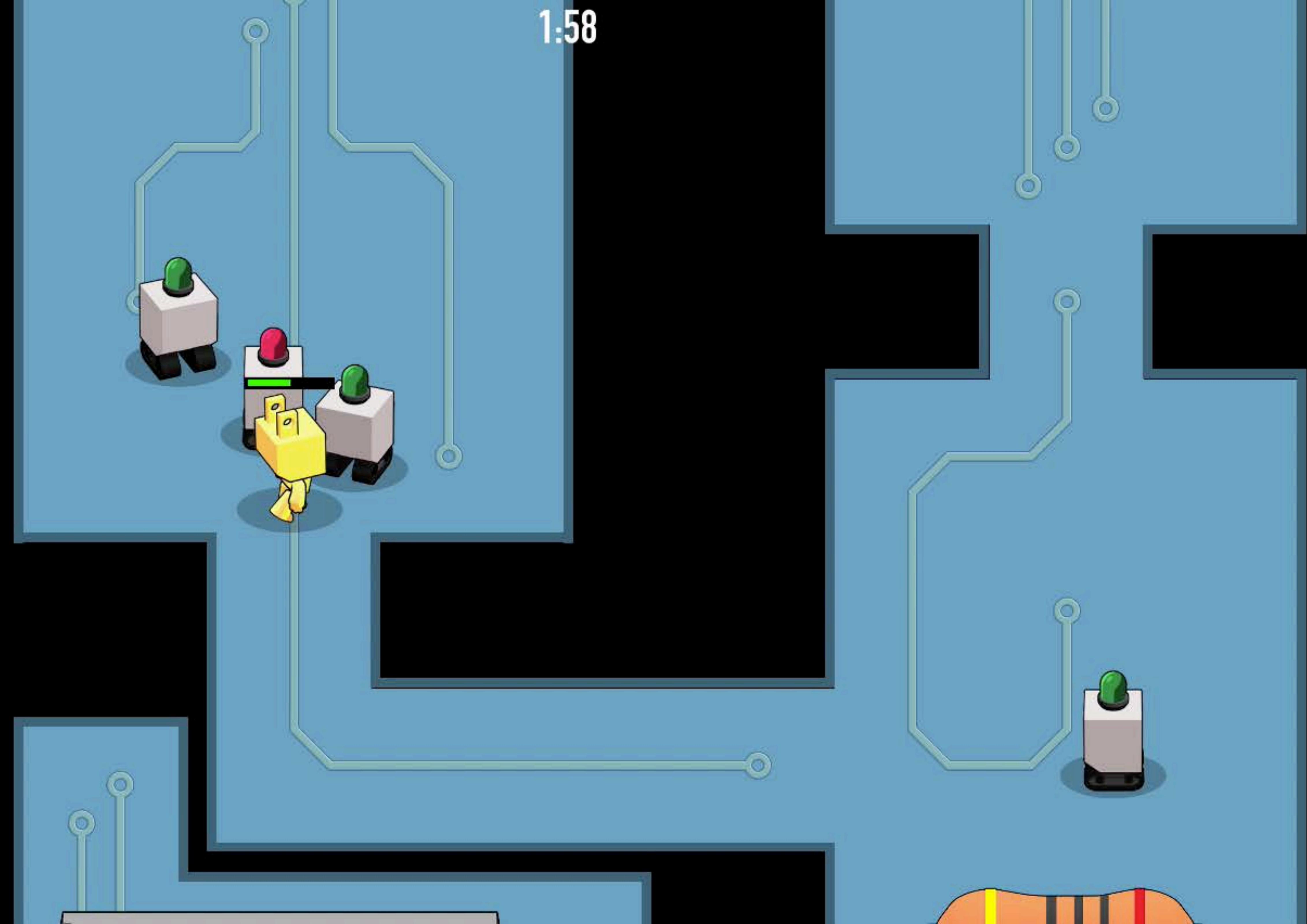
Recharging

GKStateMachine

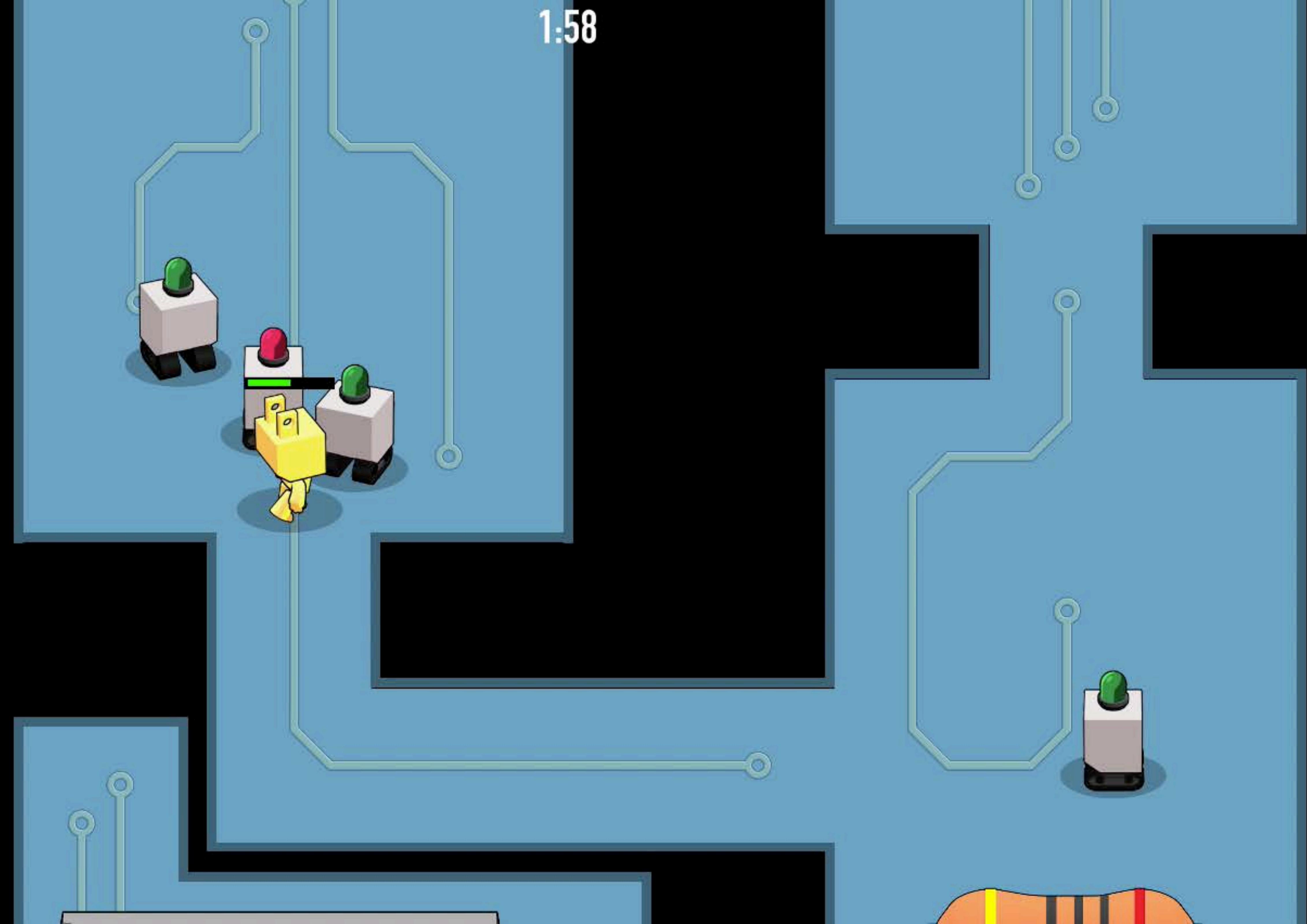
PlayerBot state



1:58

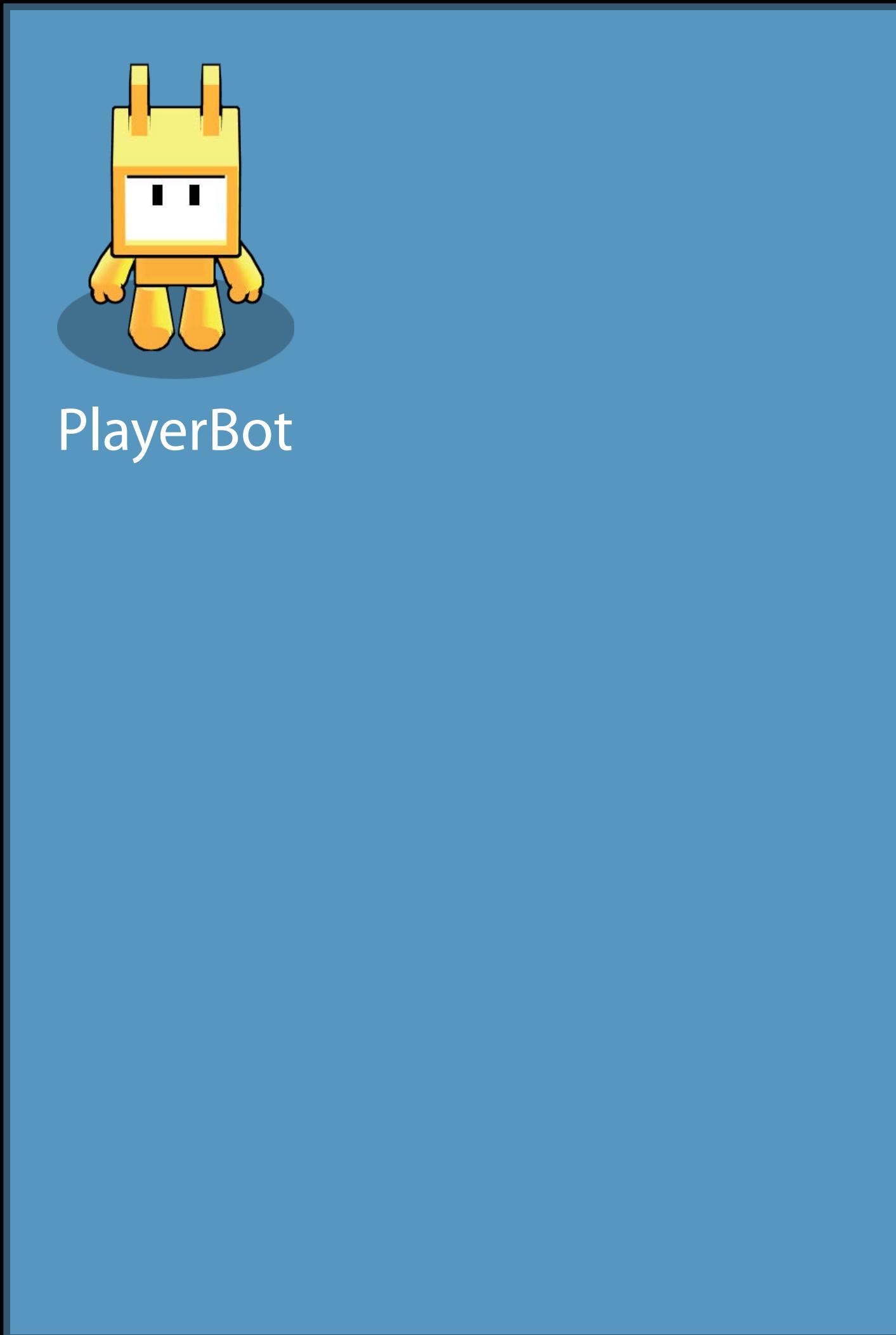


1:58

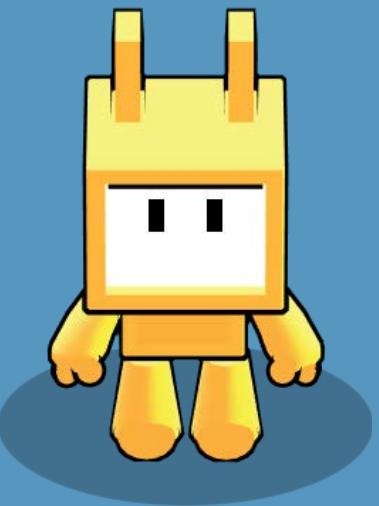


GKEntity and GKComponent

GKEntity and GKComponent



GKEntity and GKComponent

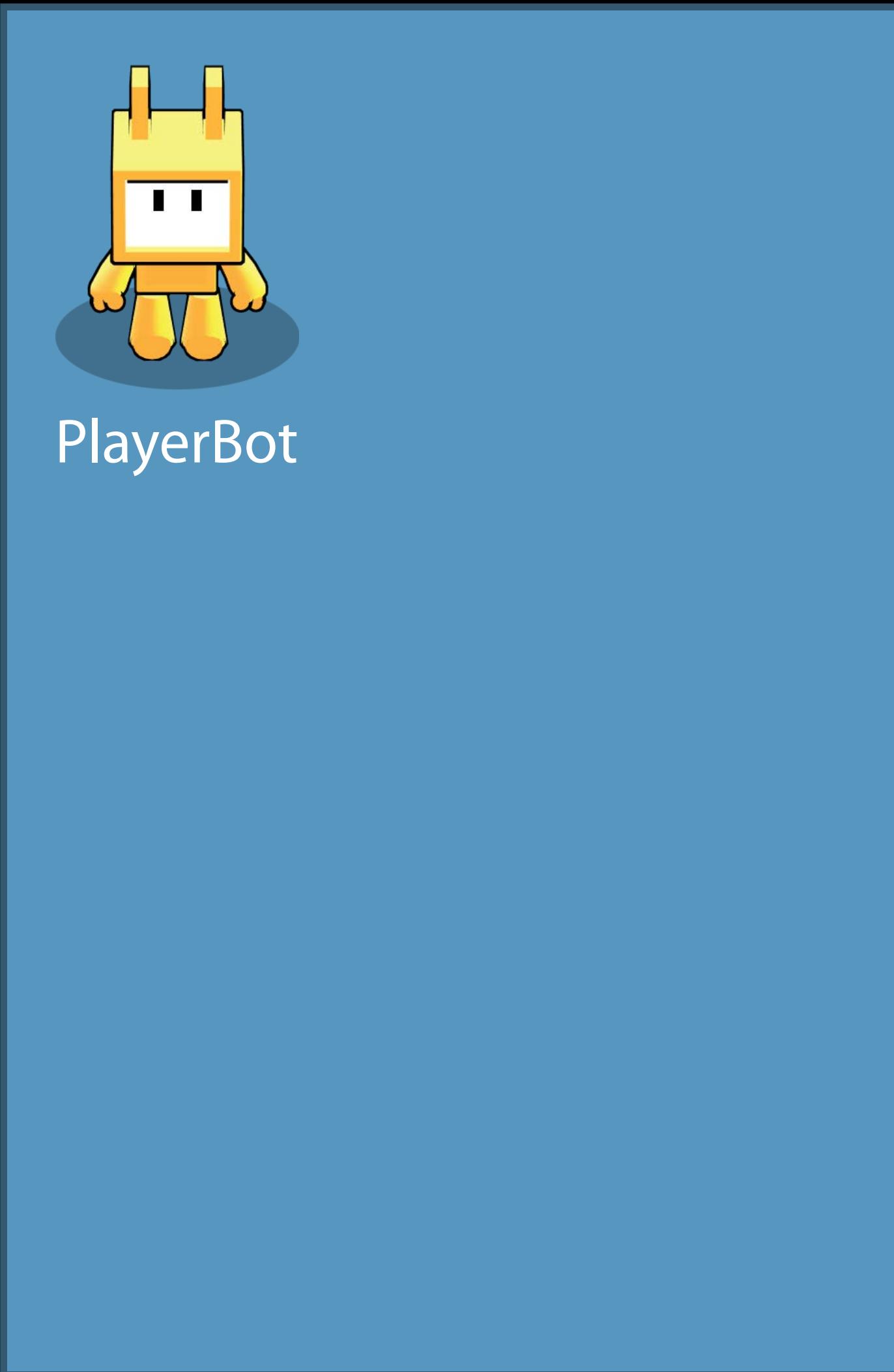


PlayerBot

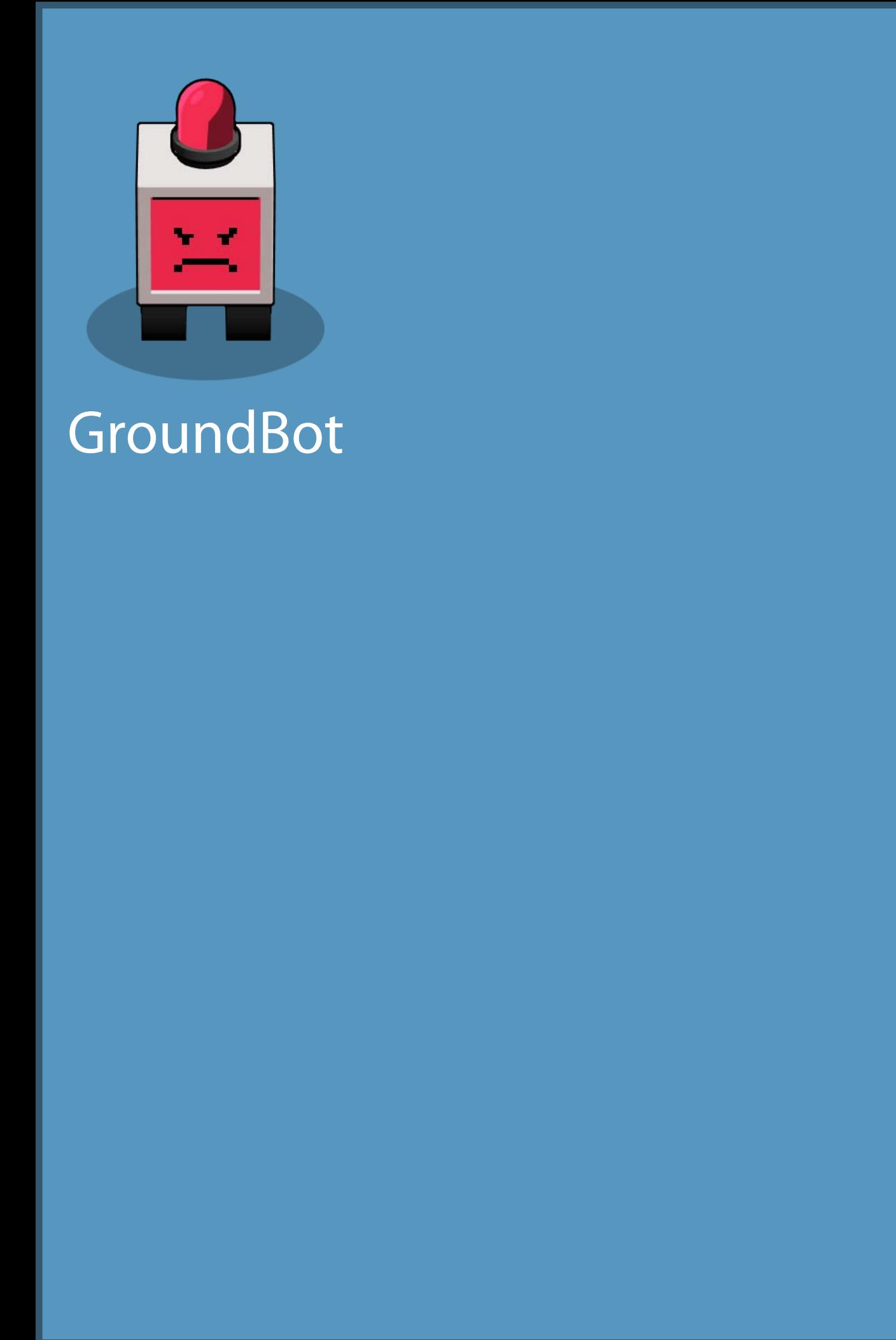


GroundBot

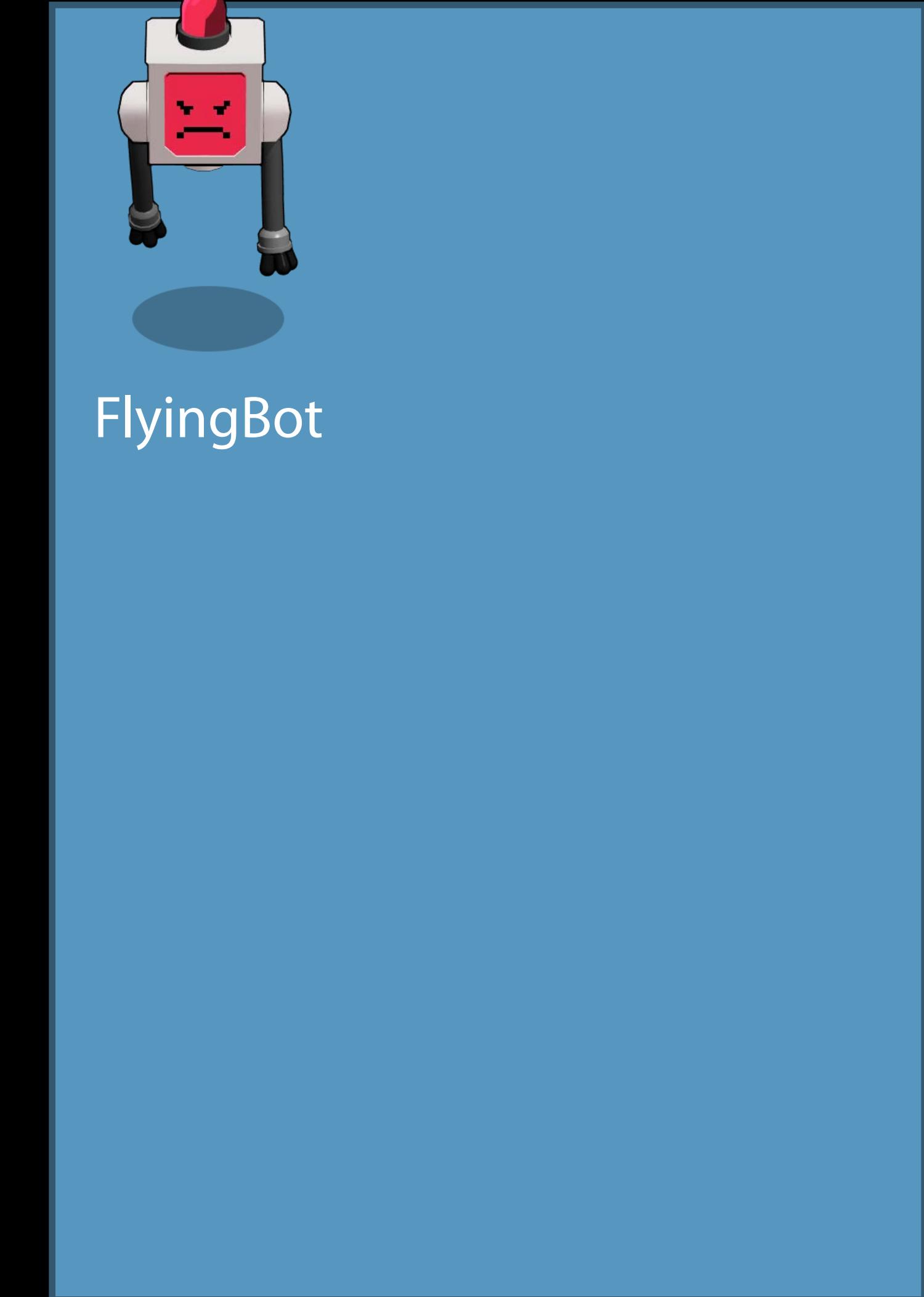
GKEntity and GKComponent



PlayerBot

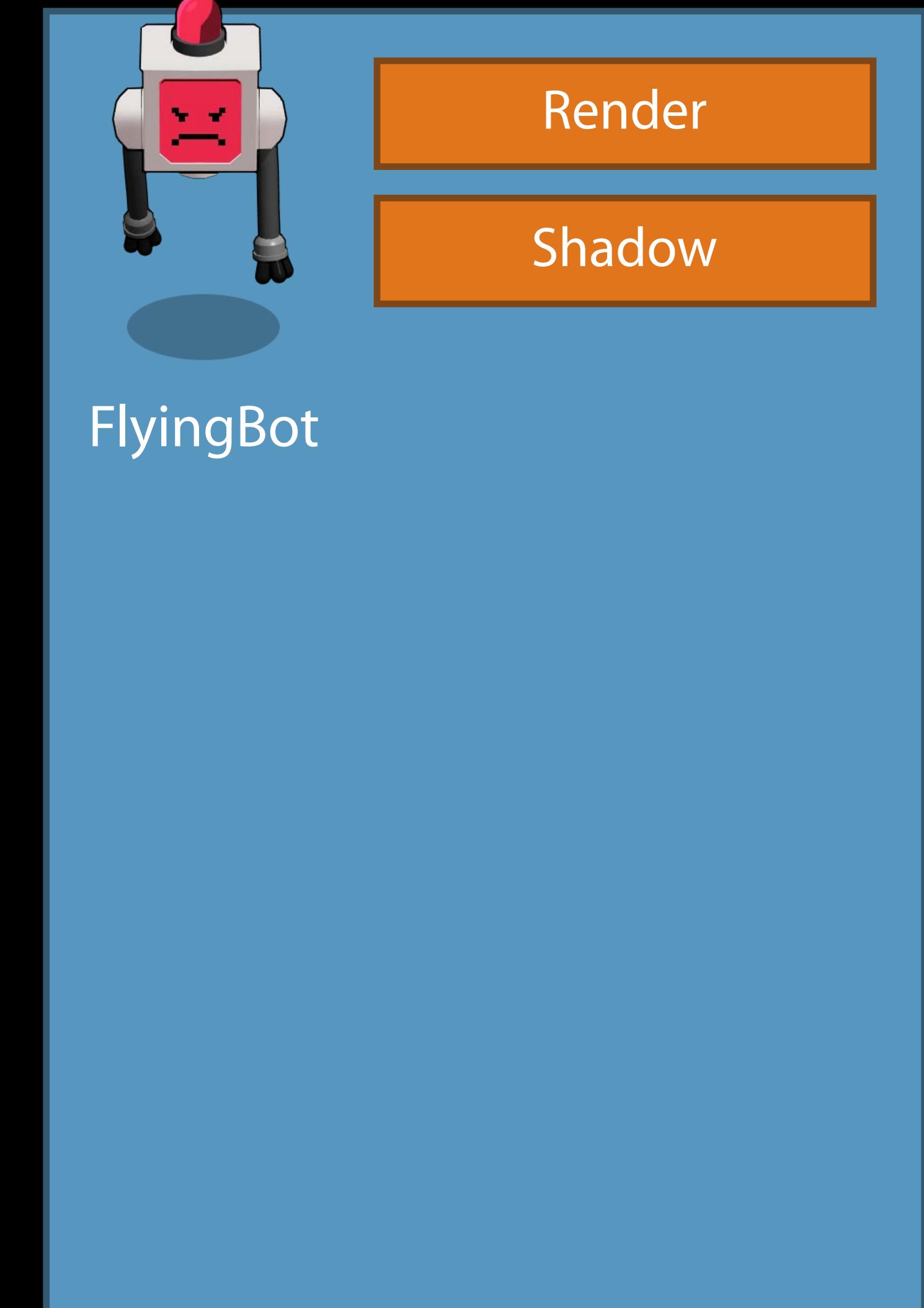
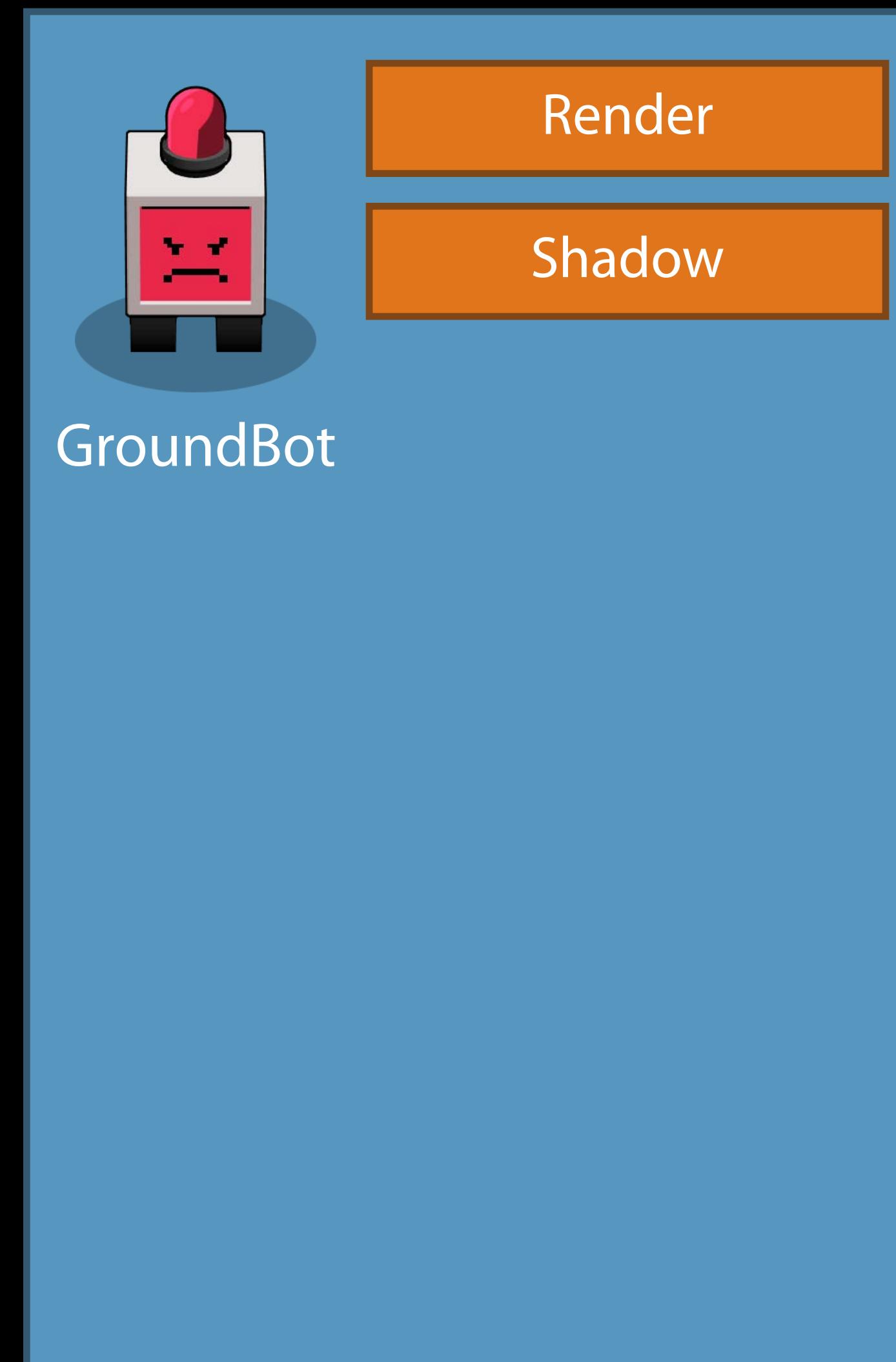
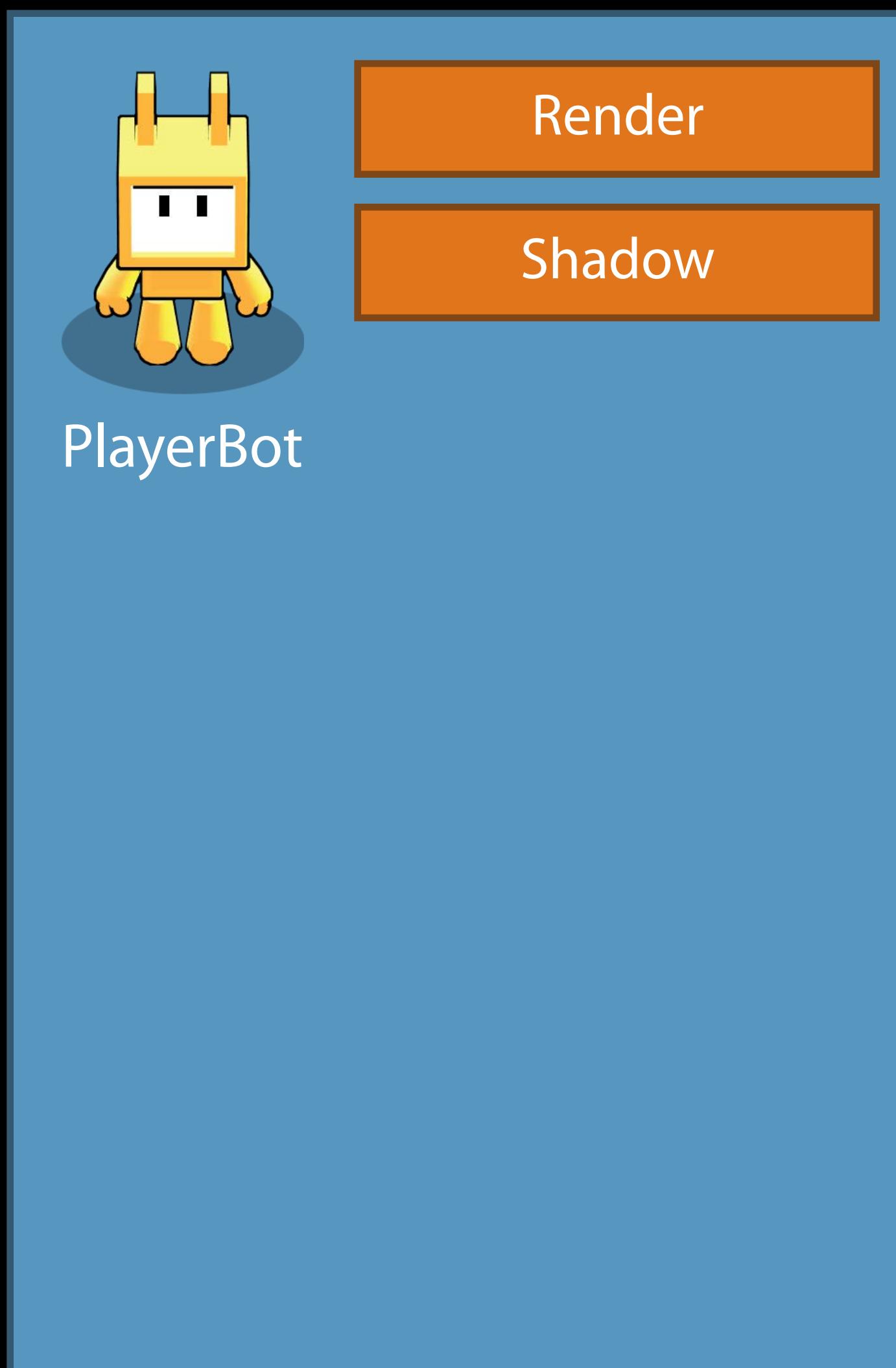


GroundBot

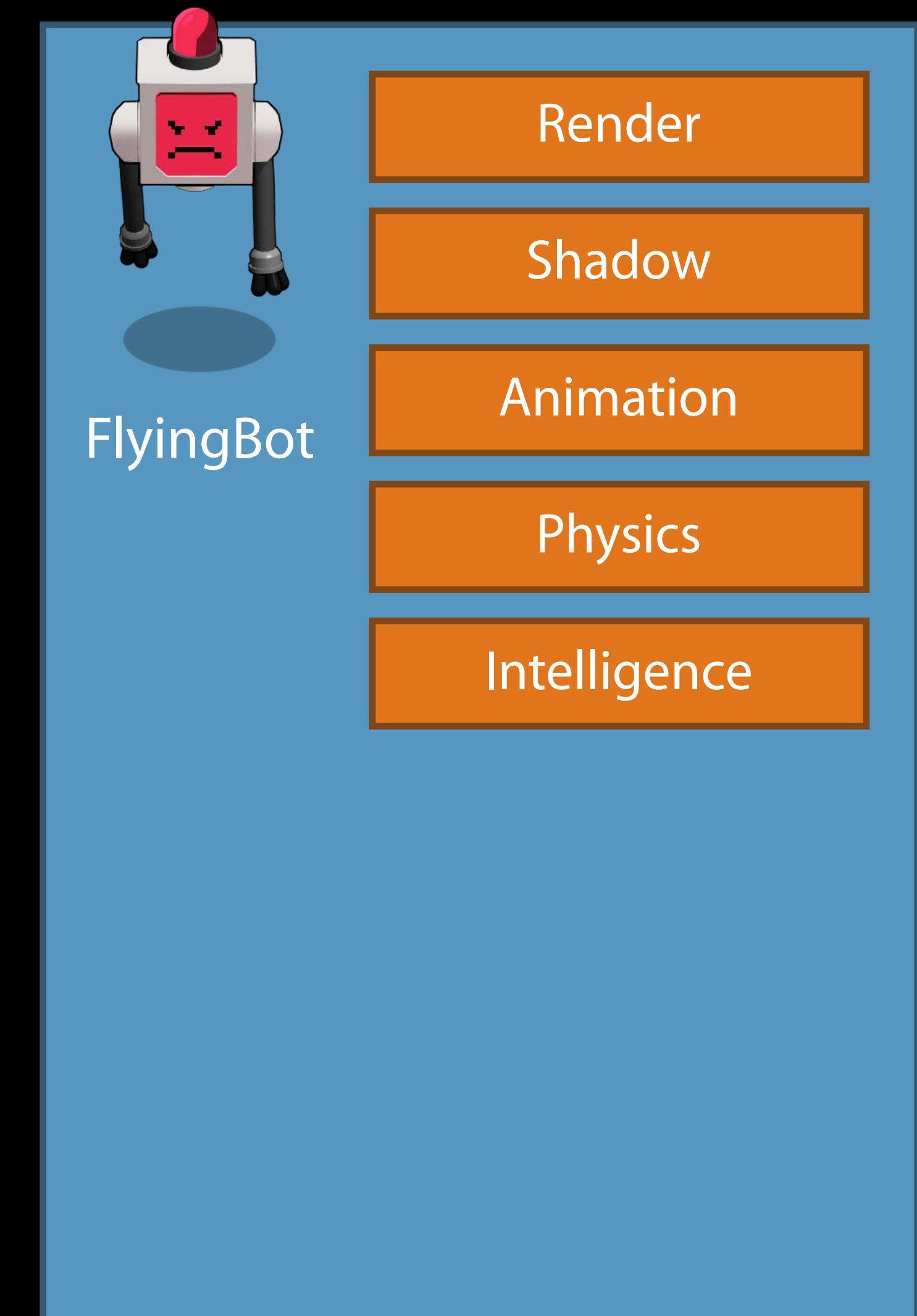
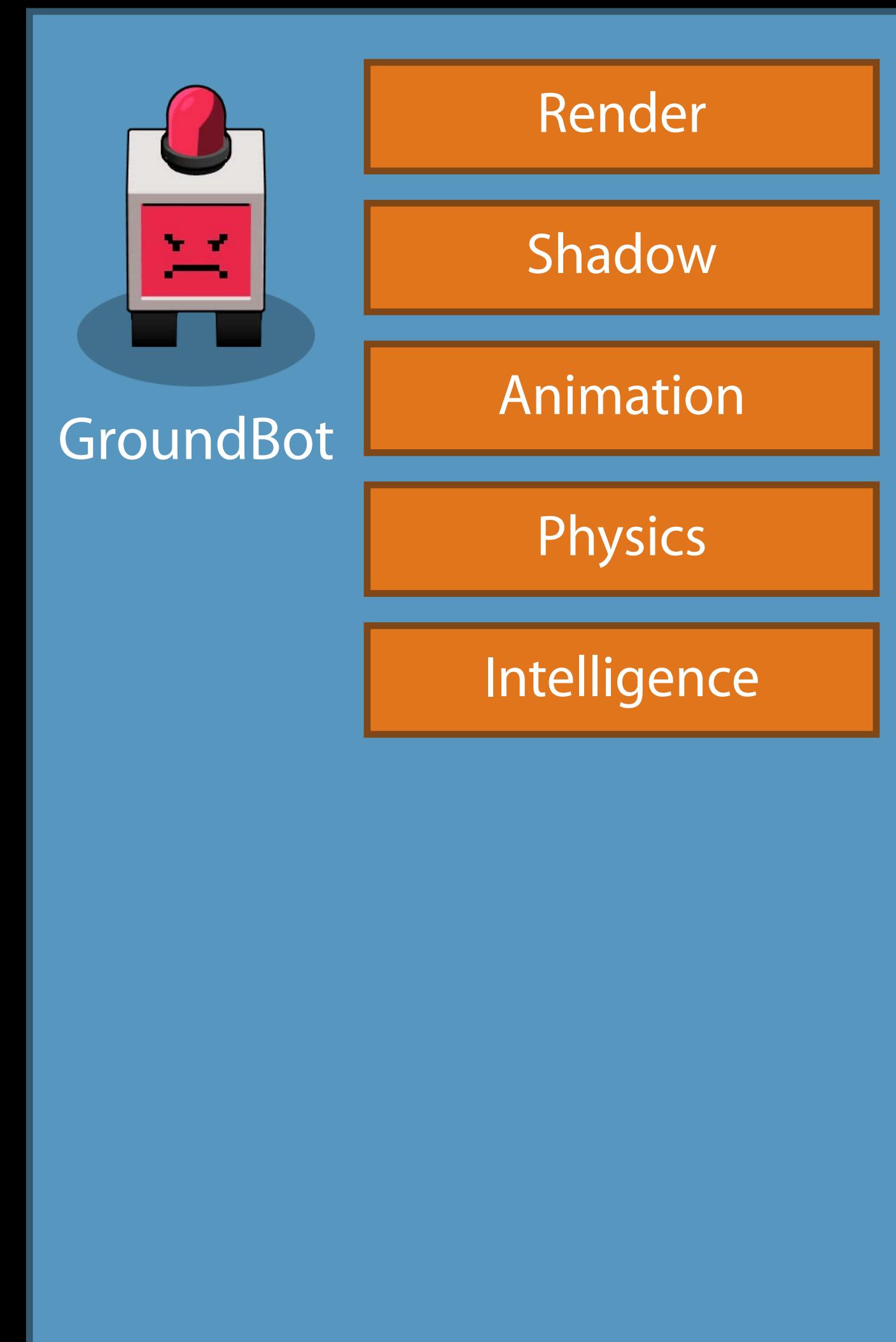
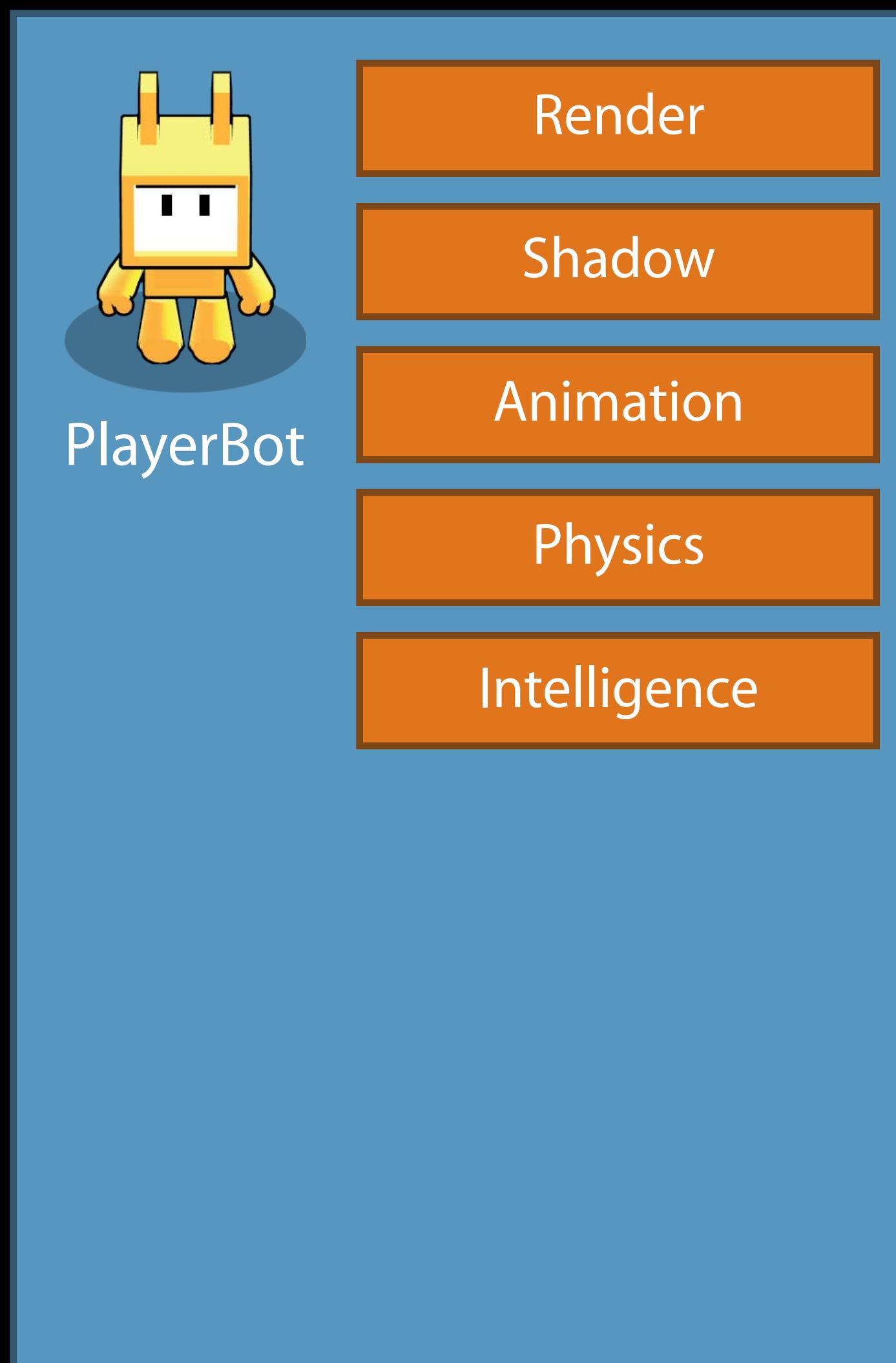


FlyingBot

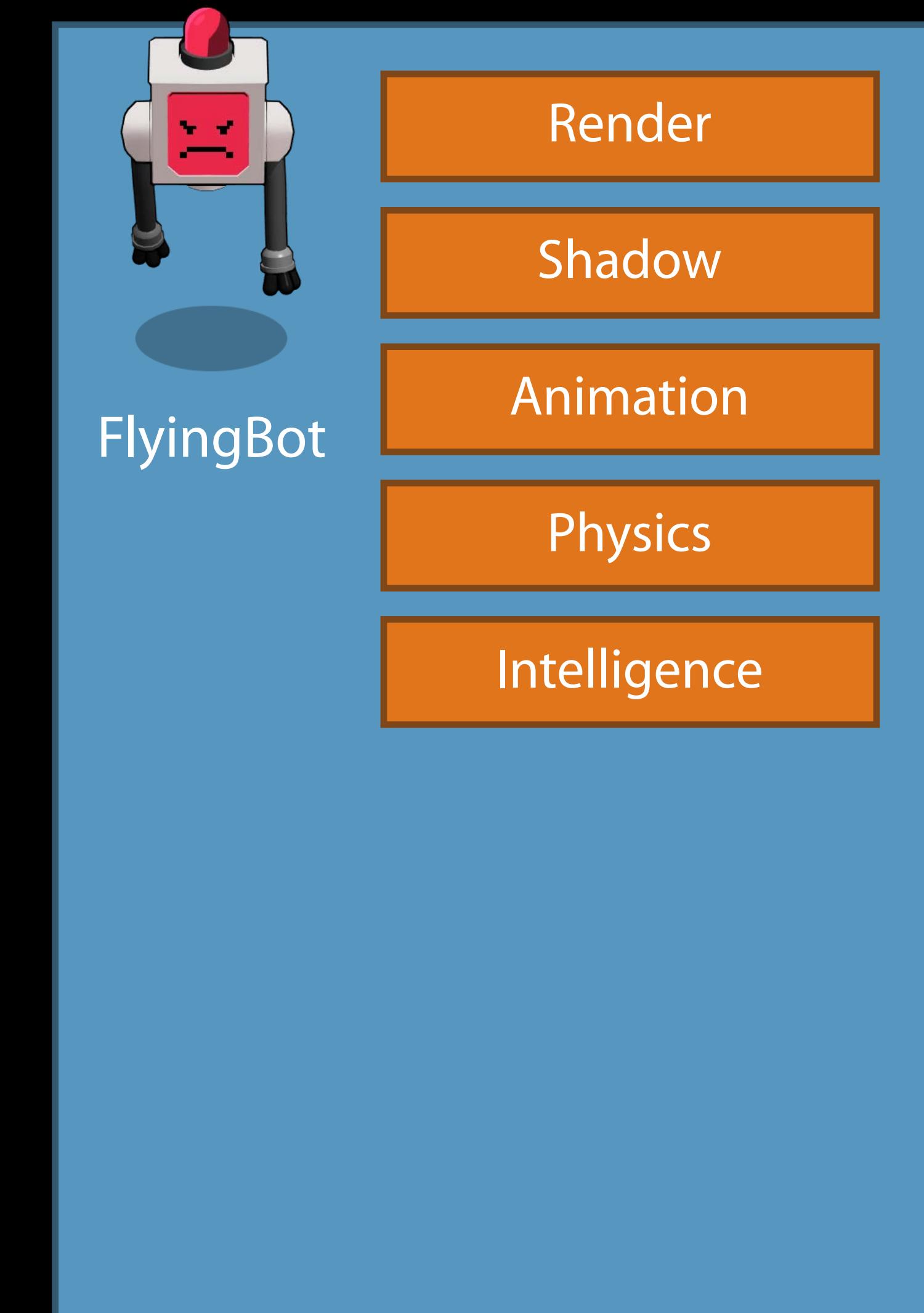
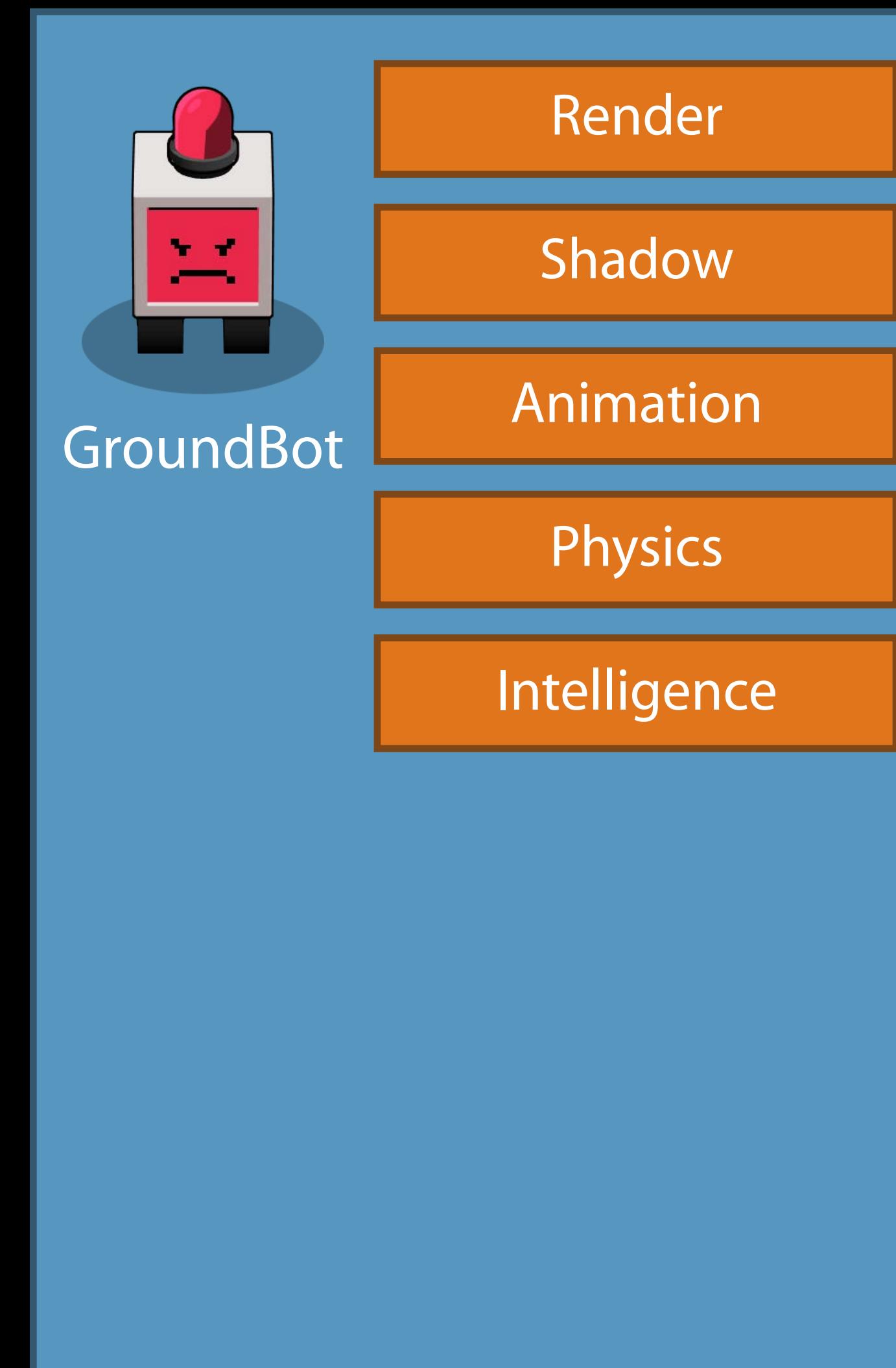
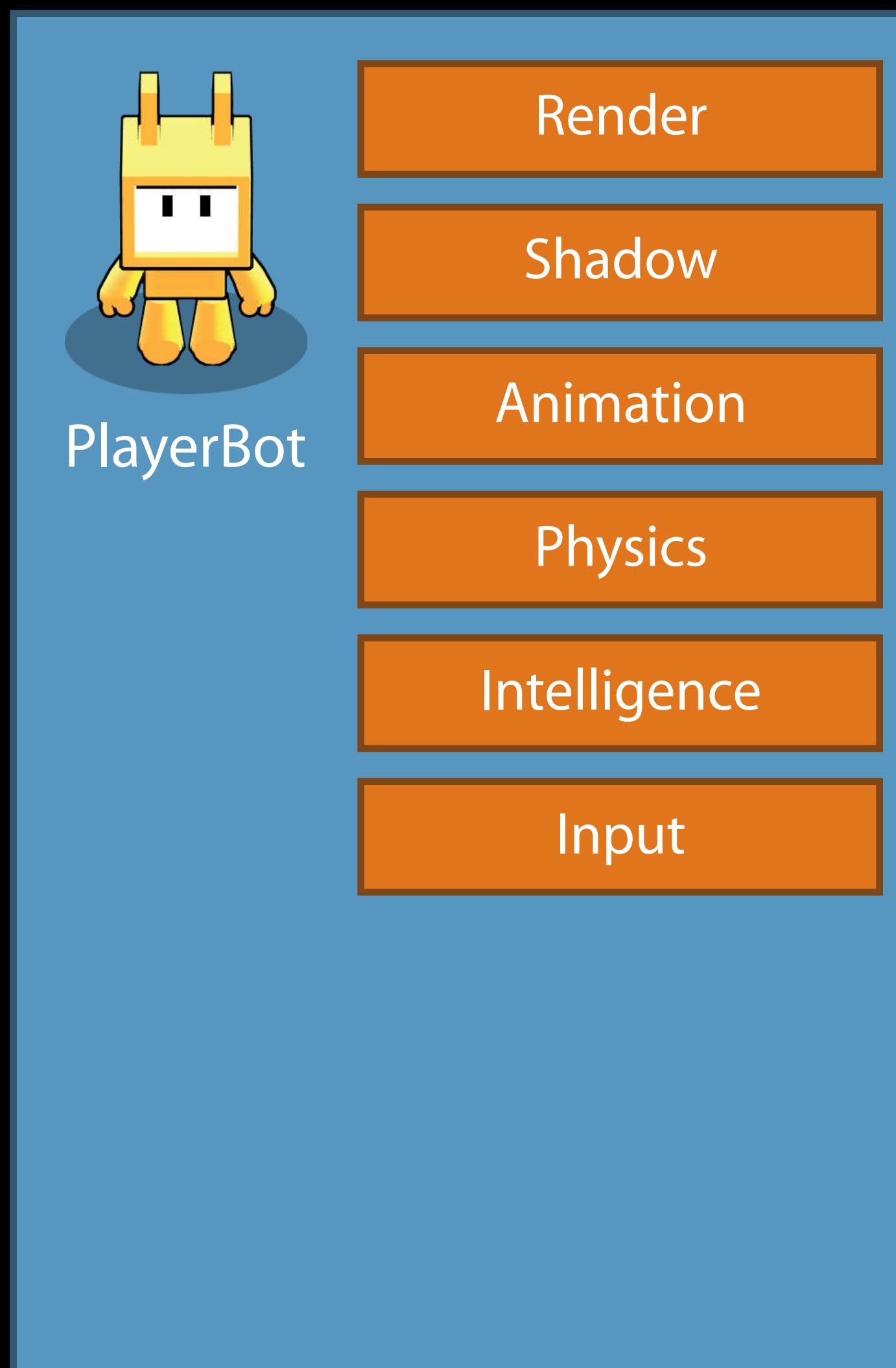
GKEntity and GKComponent



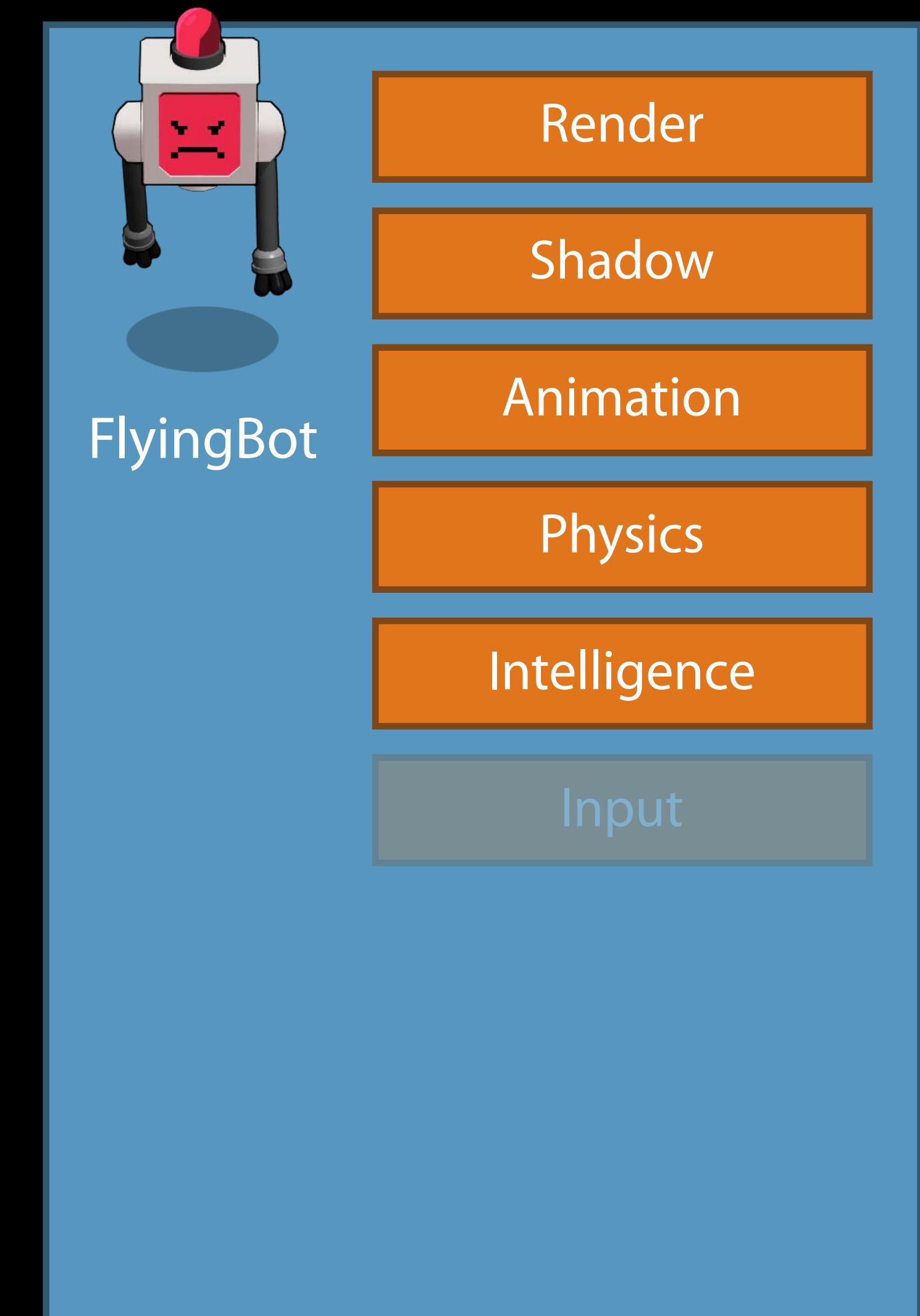
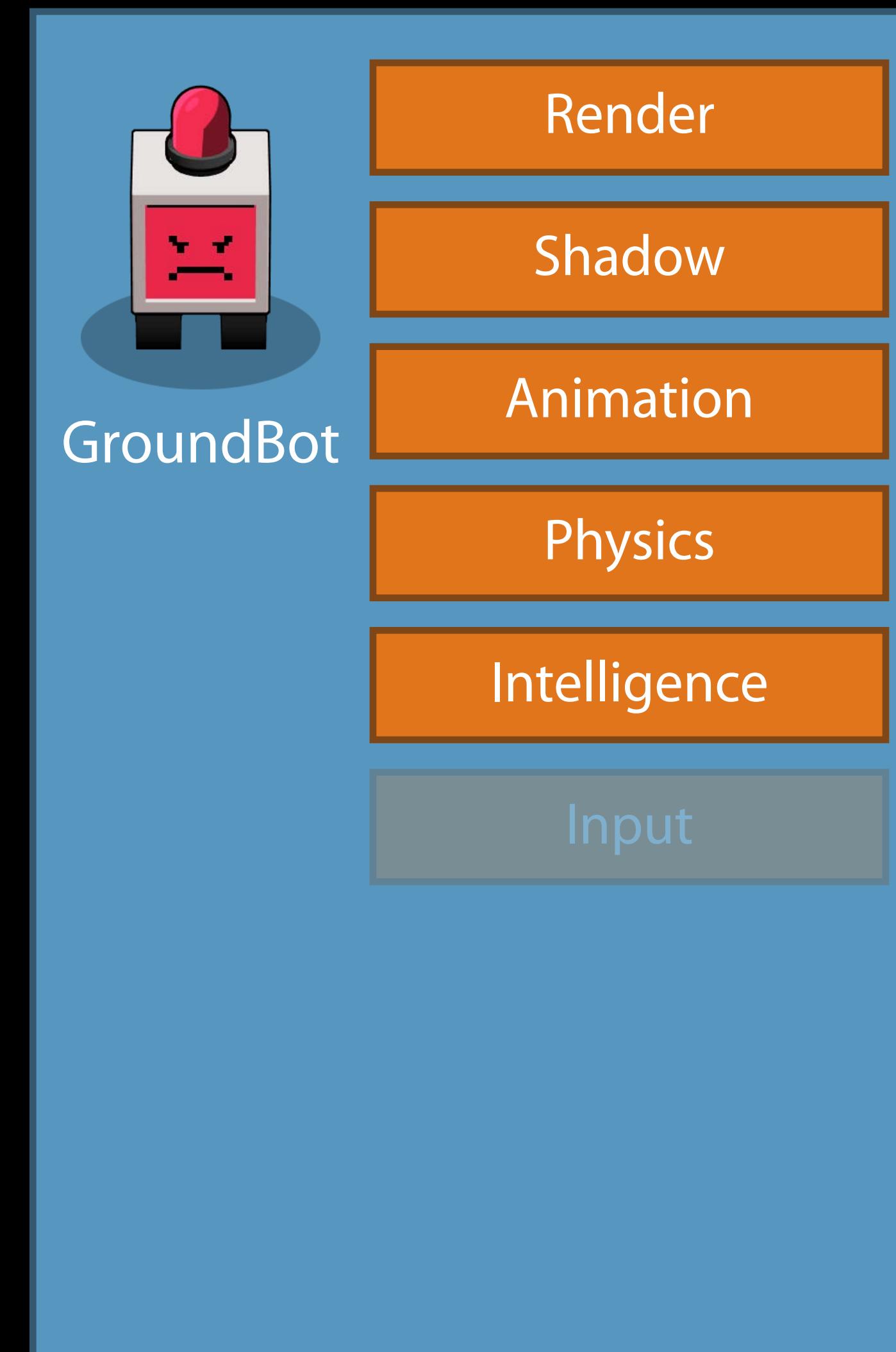
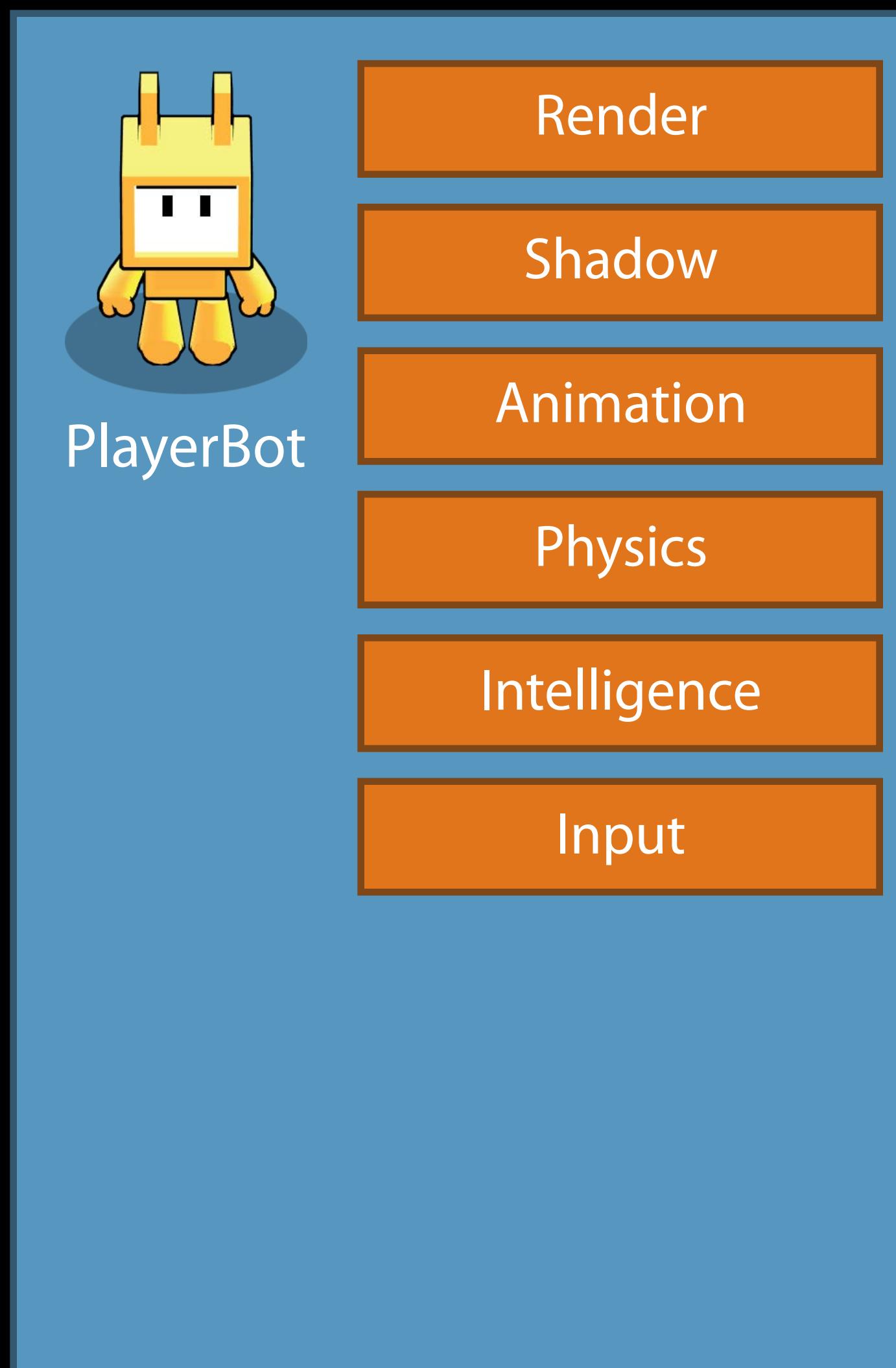
GKEntity and GKComponent



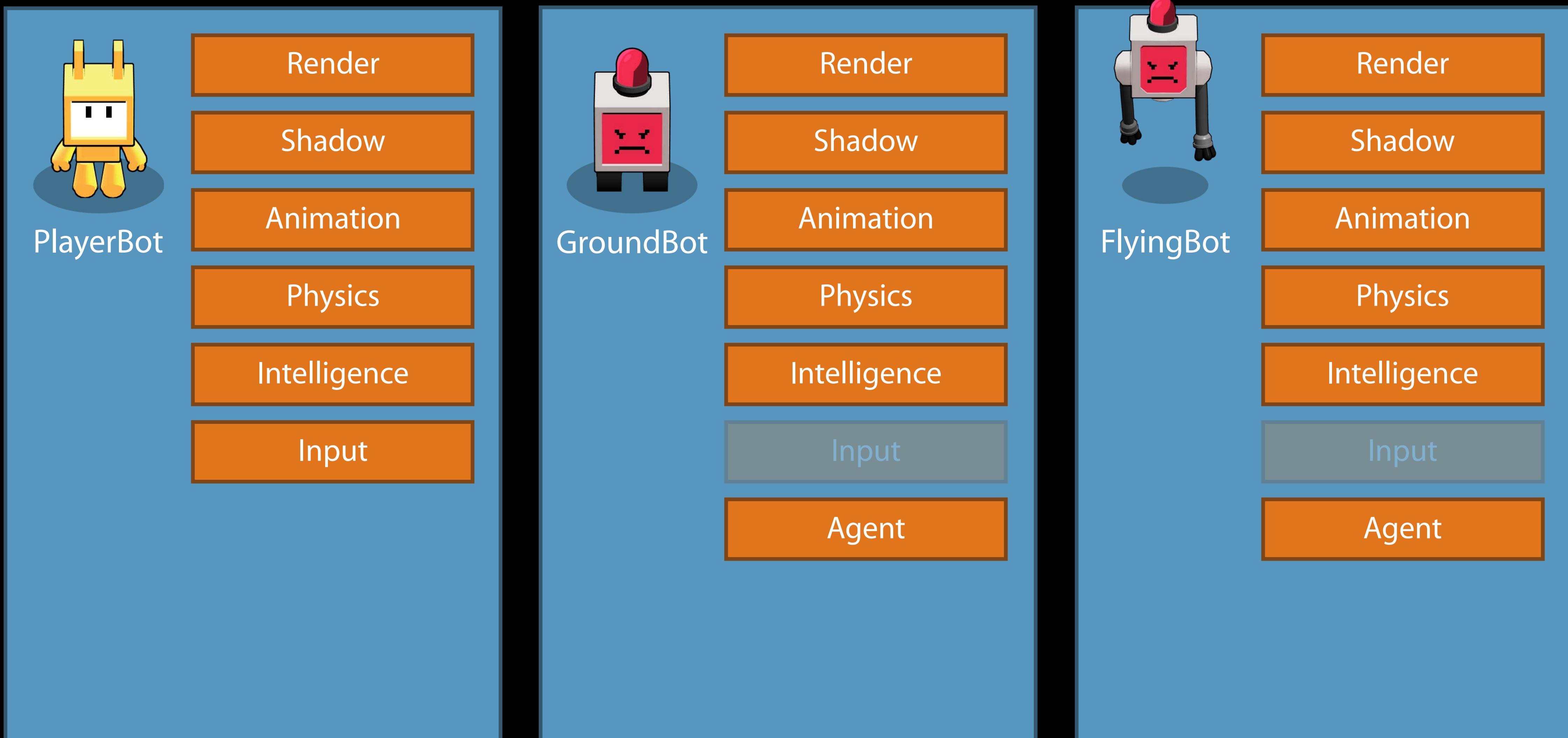
GKEntity and GKComponent



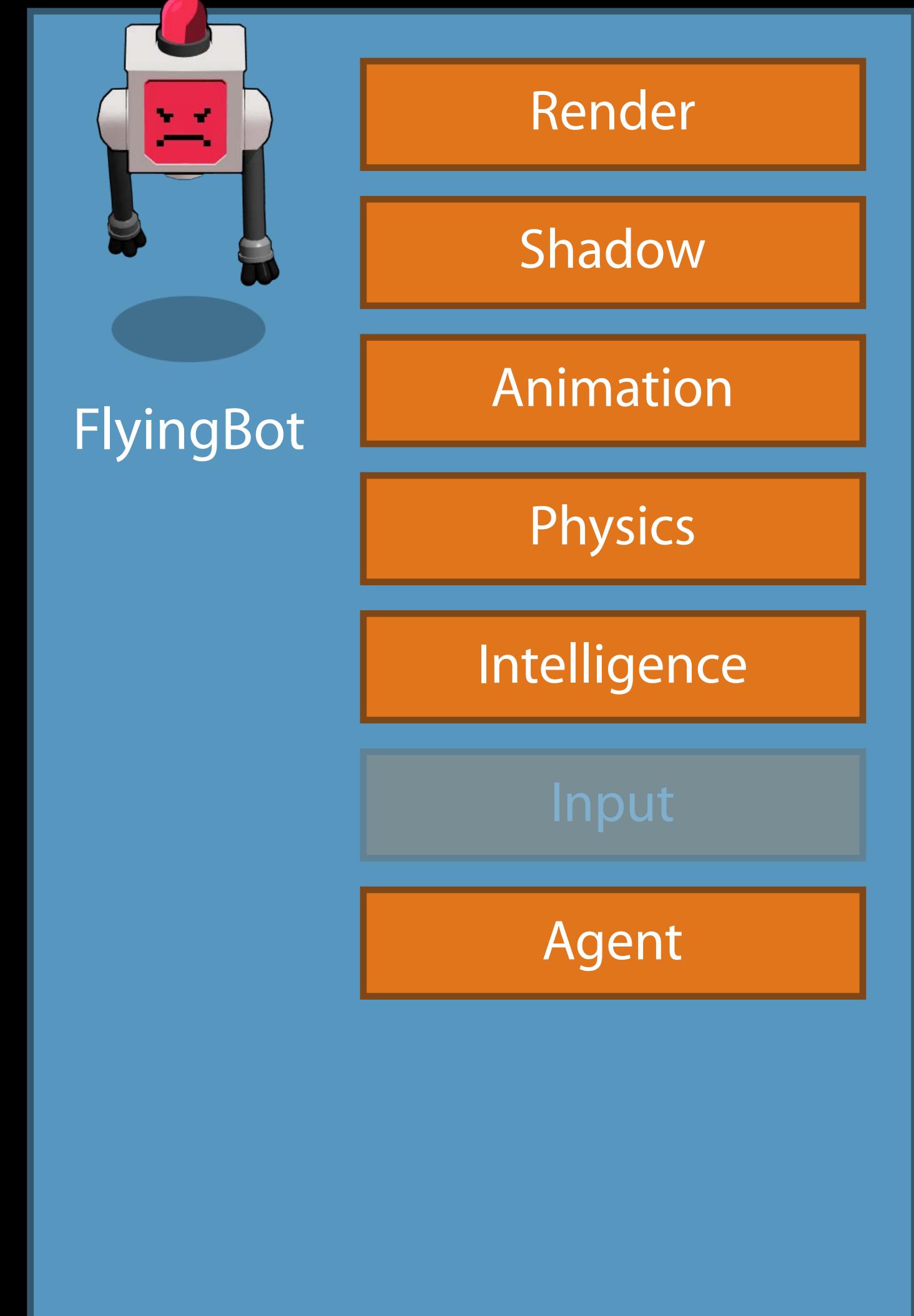
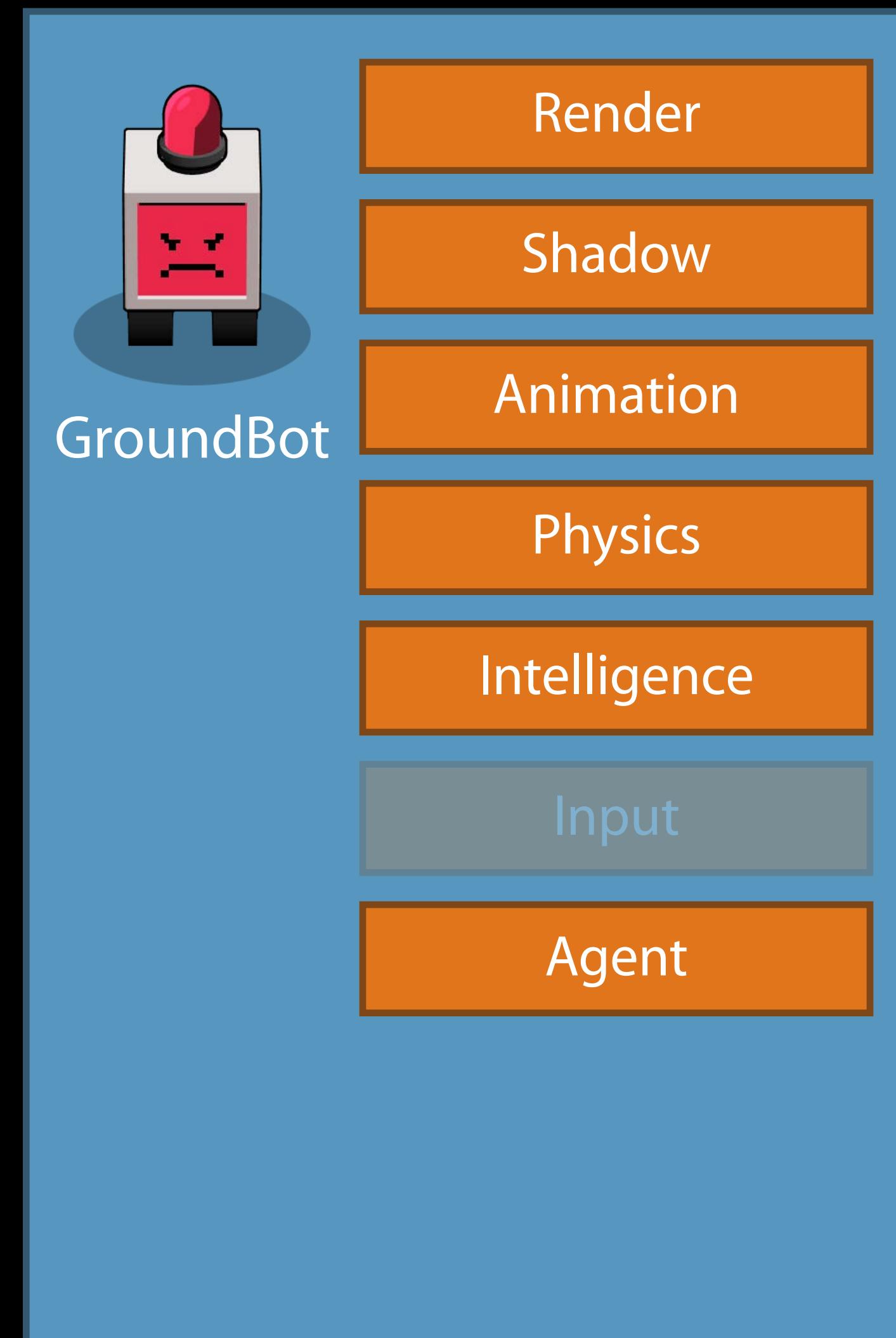
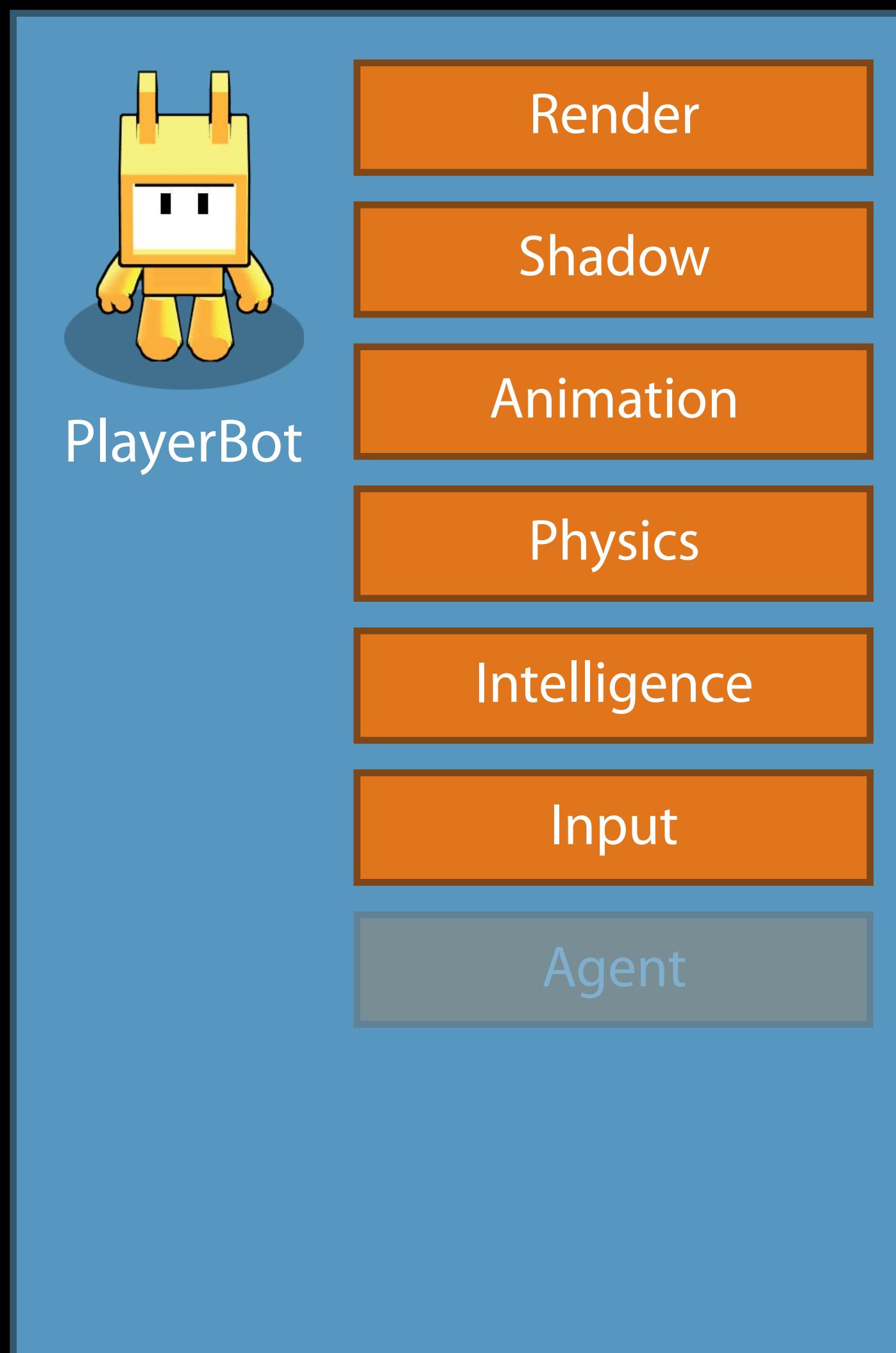
GKEntity and GKComponent



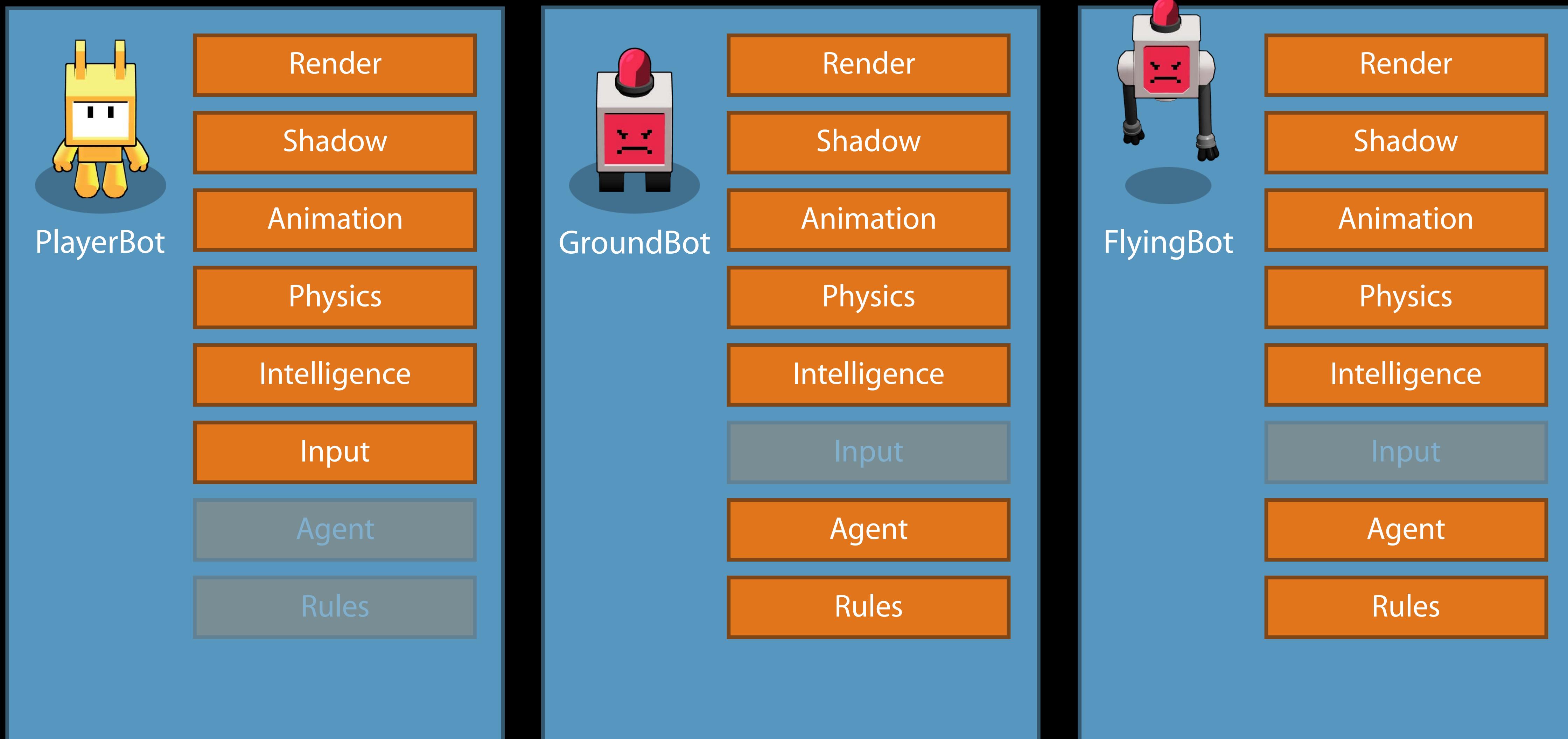
GKEntity and GKComponent



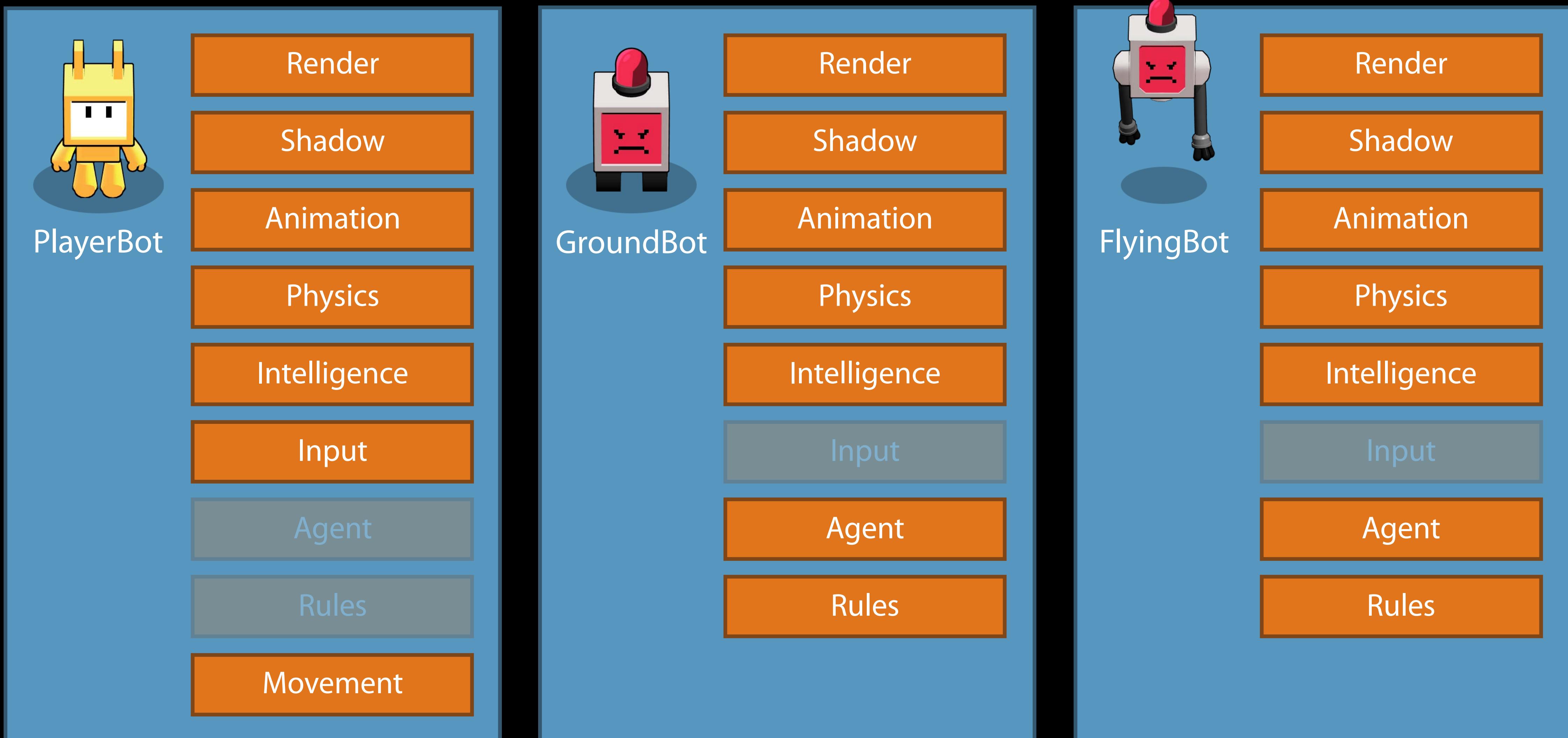
GKEntity and GKComponent



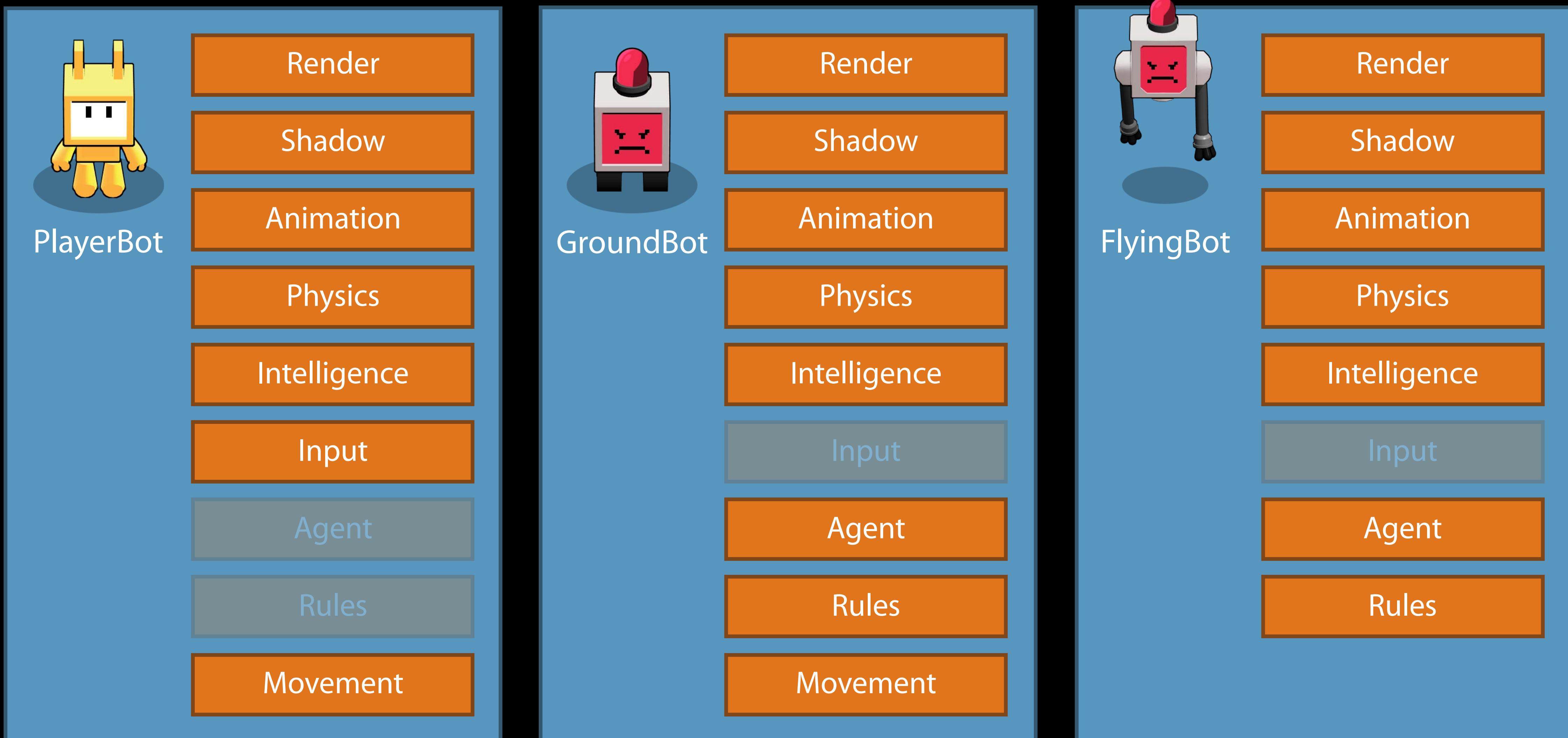
GKEntity and GKComponent



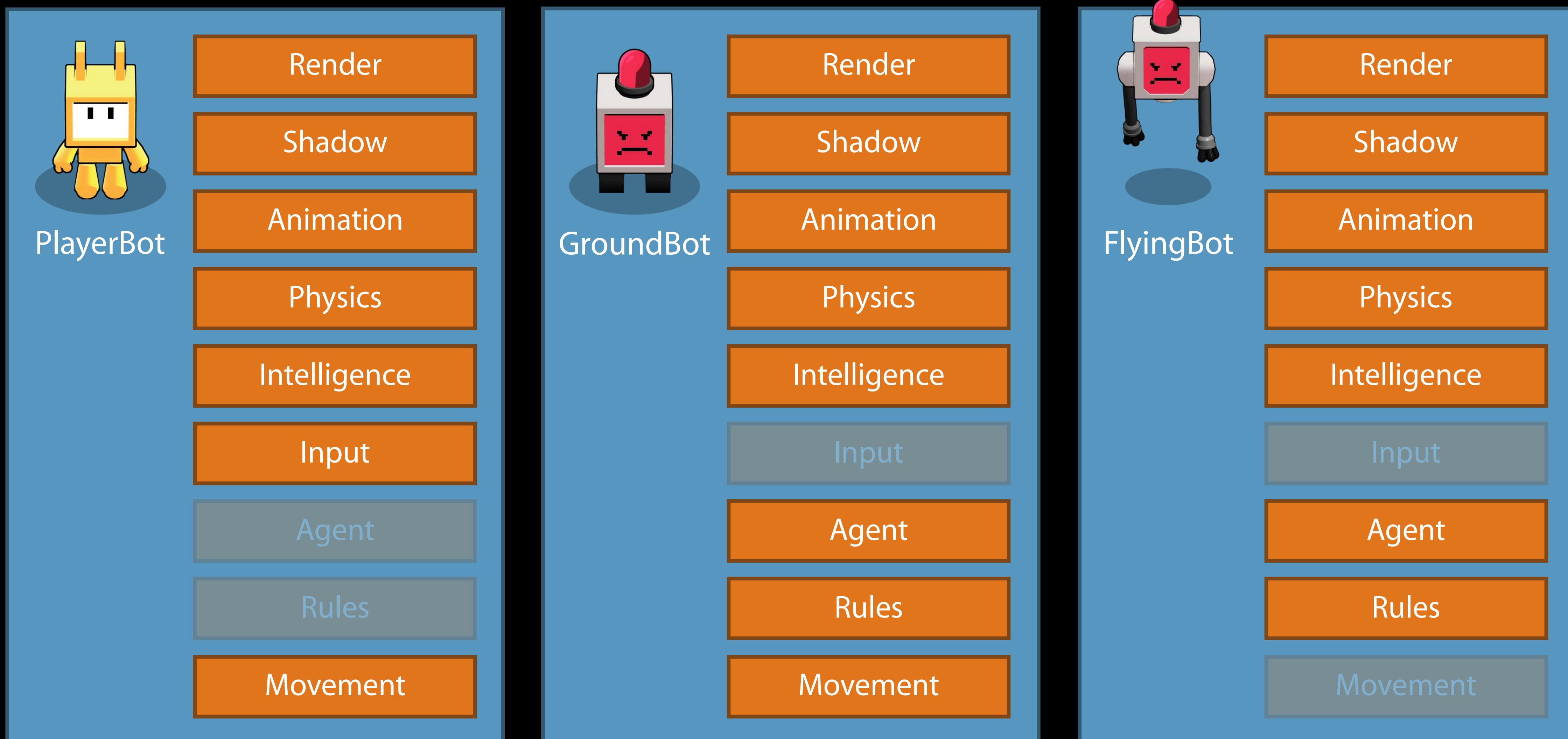
GKEntity and GKComponent



GKEntity and GKComponent



GKEntity and GKComponent



Logic and Gameplay

Dave Schaefgen Sample Code Engineer

TaskBot Intelligence

What's my motivation?

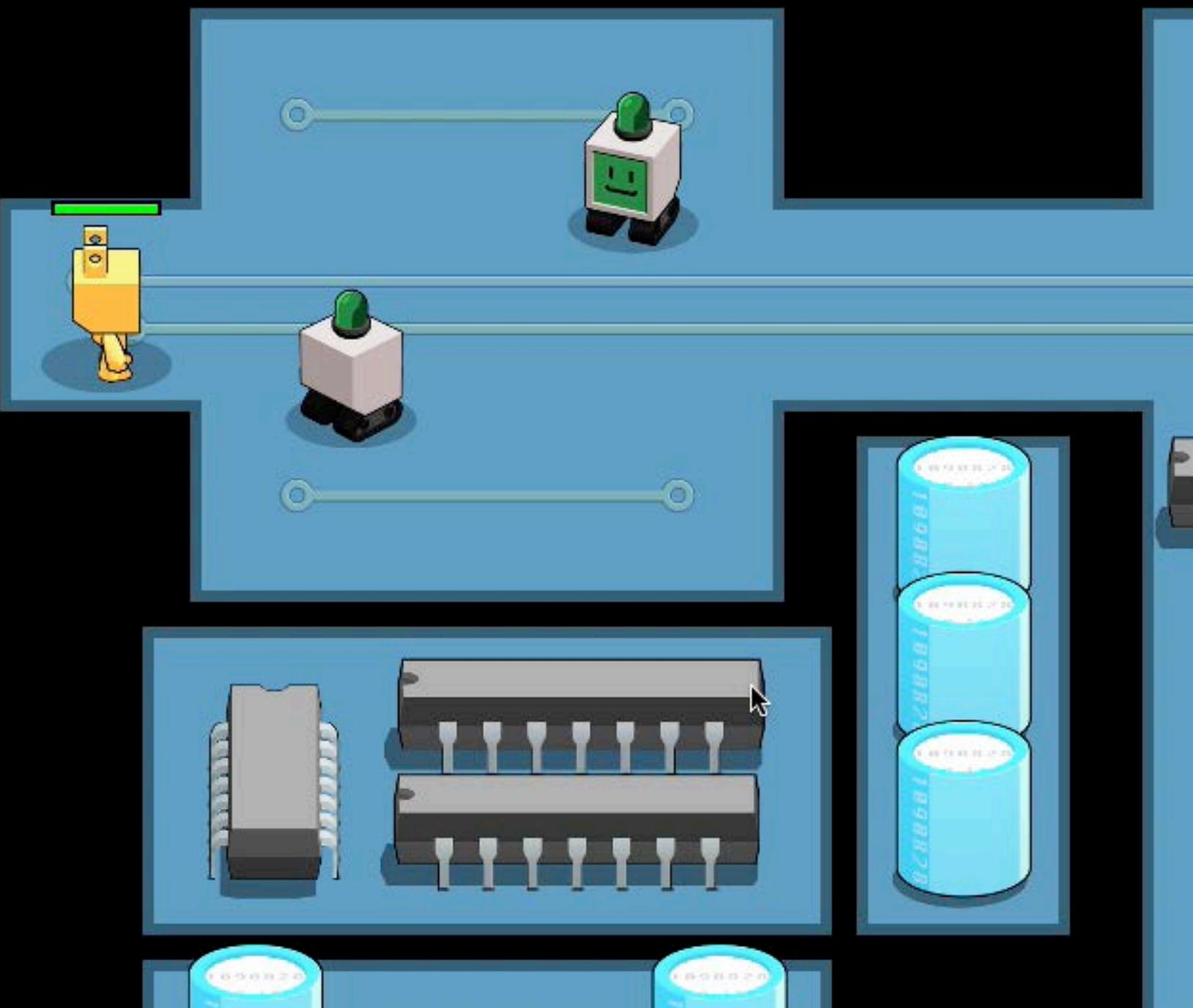
TaskBot Intelligence

What's my motivation?

Good

2:57

- Keep things working



TaskBot Intelligence

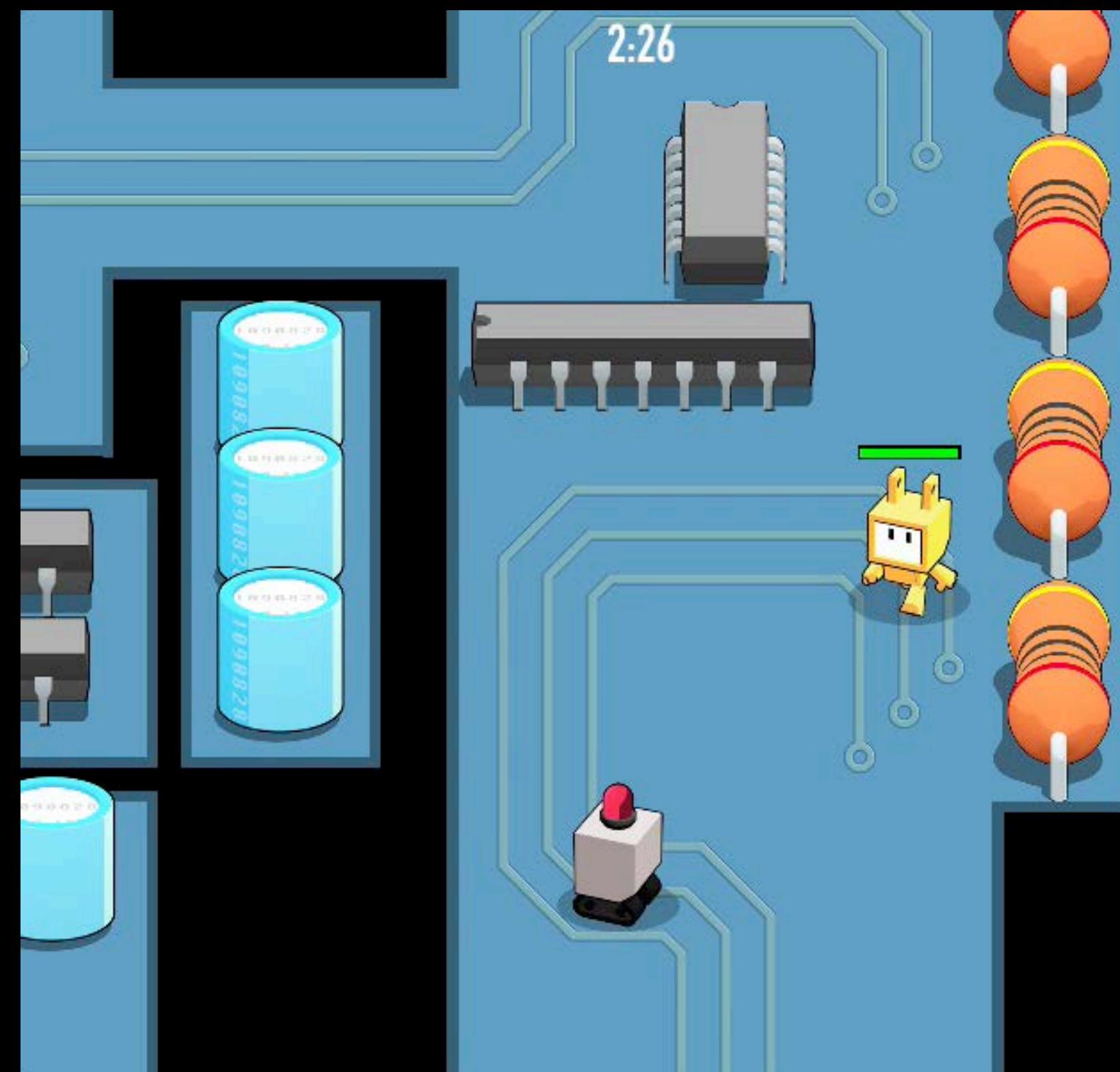
What's my motivation?

Good

- Keep things working

Bad

- Attack the PlayerBot



TaskBot Intelligence

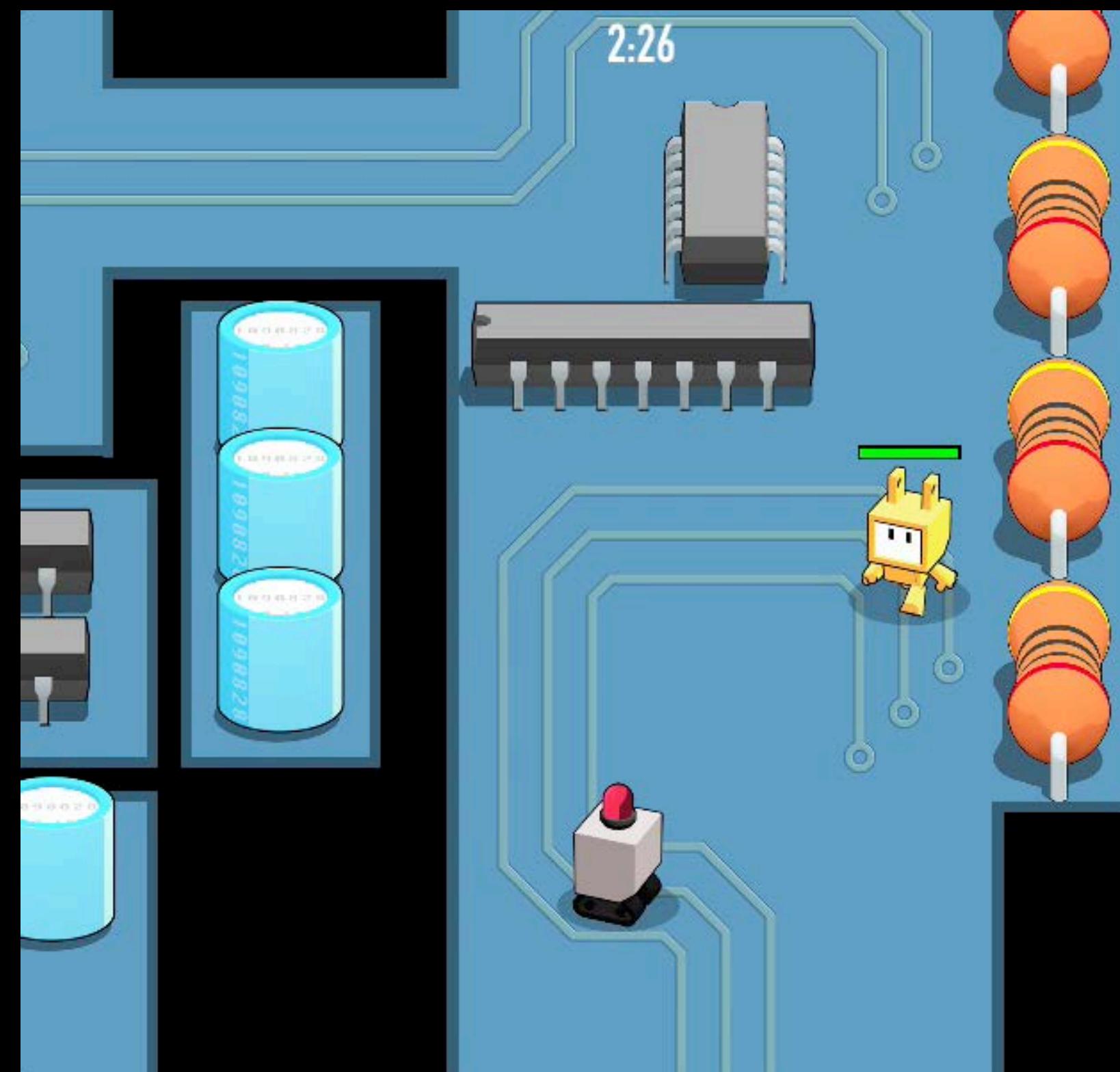
What's my motivation?

Good

- Keep things working

Bad

- Attack the PlayerBot



TaskBot Intelligence

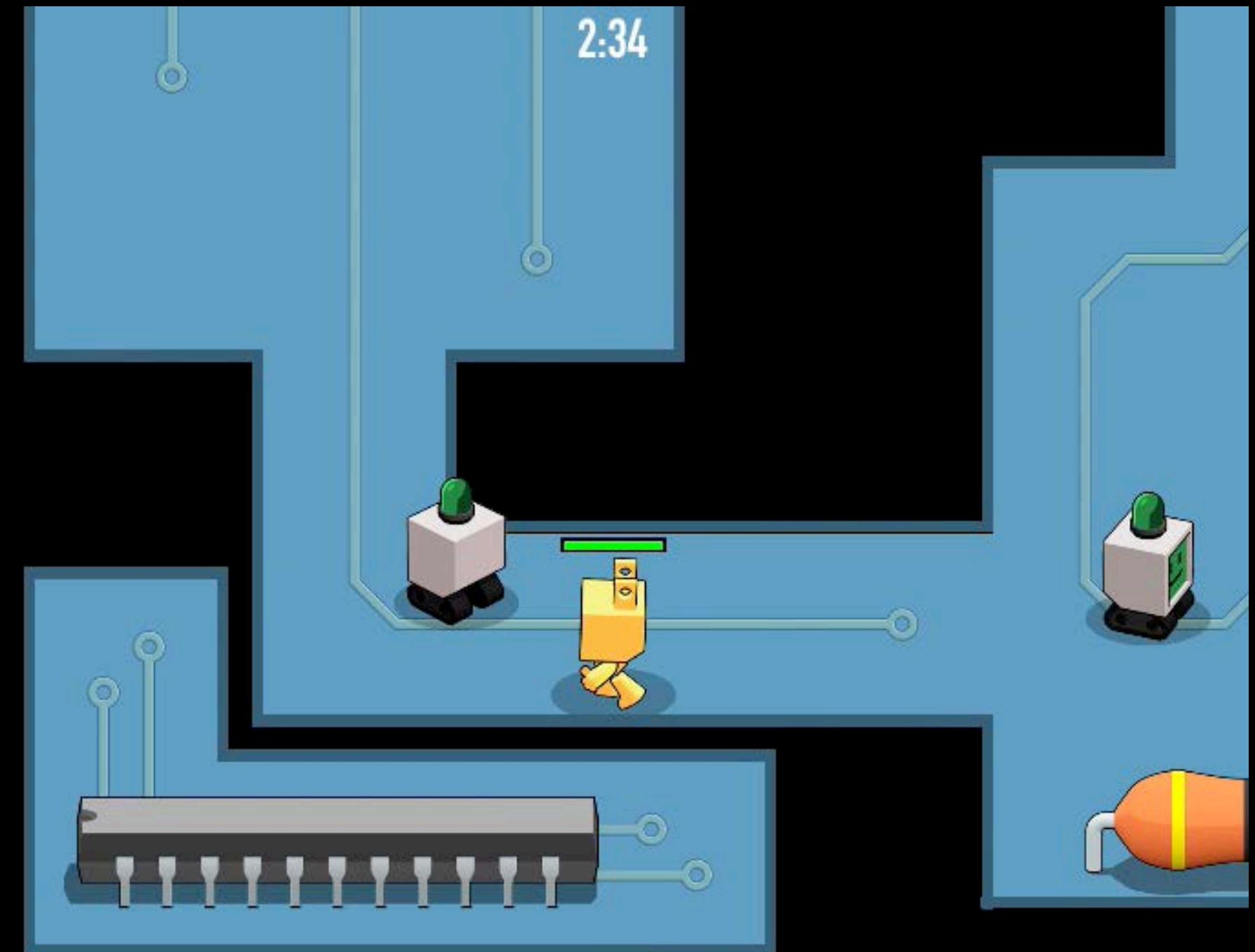
What's my motivation?

Good

- Keep things working

Bad

- Attack the PlayerBot
- Turn other TaskBots



TaskBot Intelligence

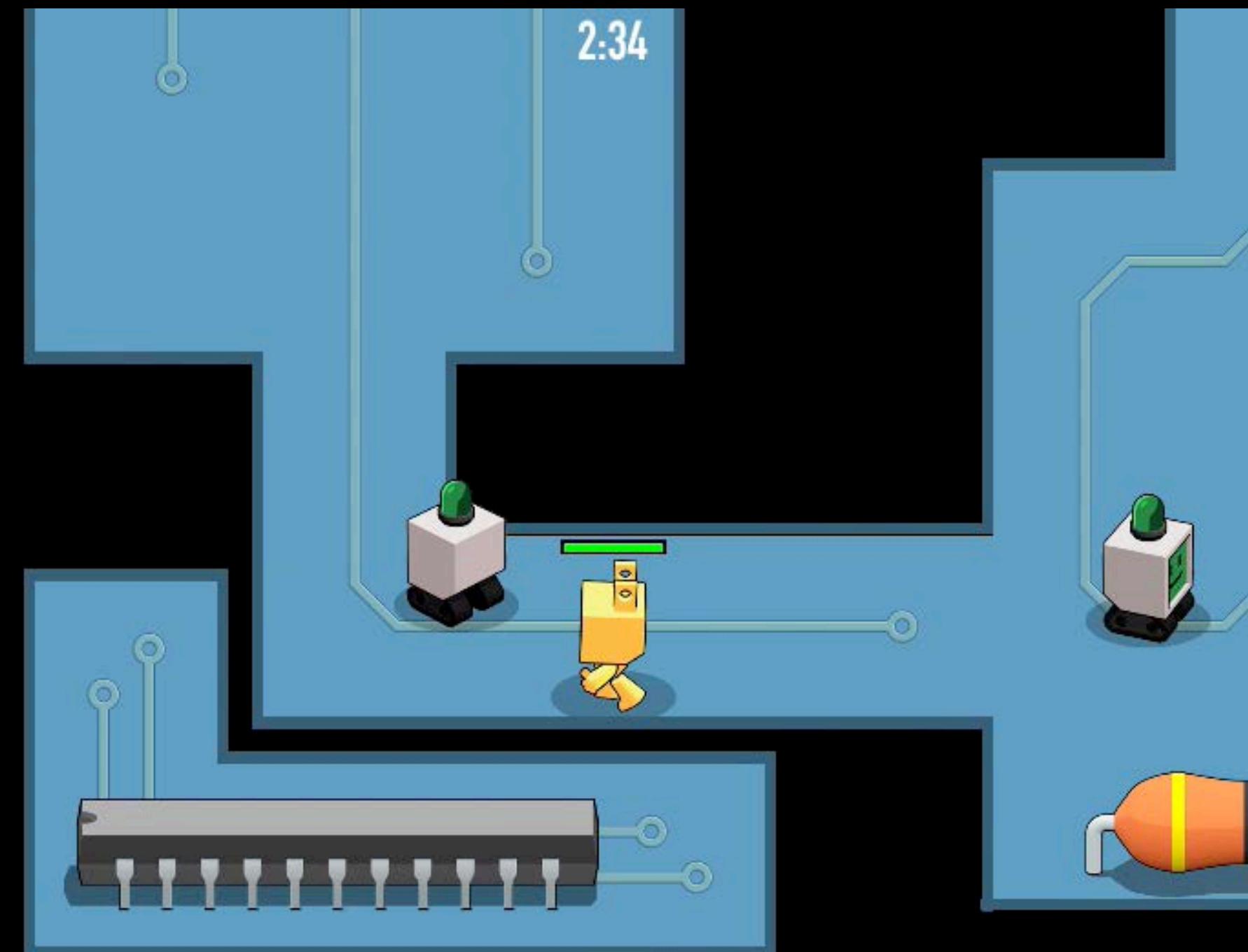
What's my motivation?

Good

- Keep things working

Bad

- Attack the PlayerBot
- Turn other TaskBots



TaskBot Intelligence

What's my motivation?

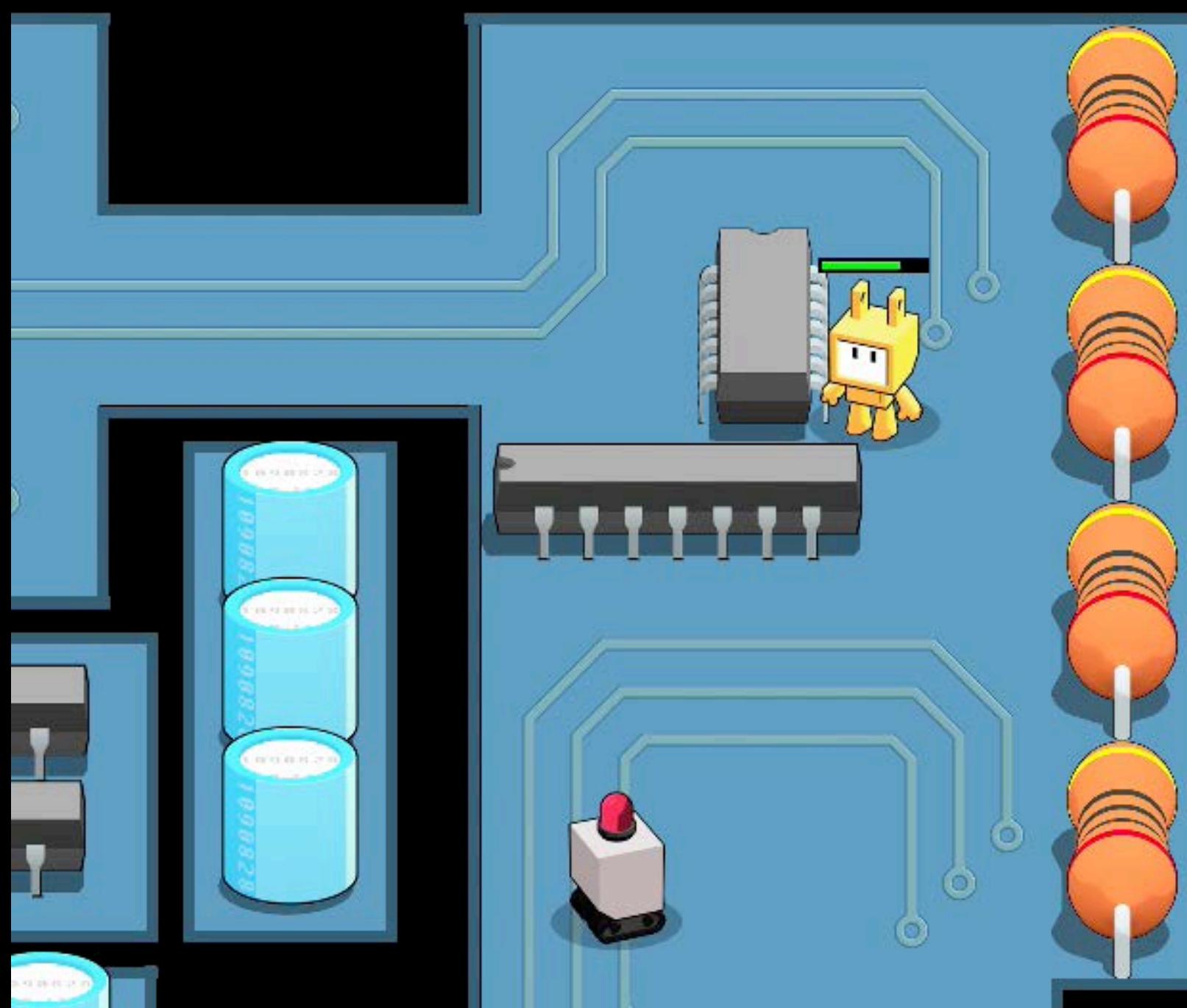
Good

- Keep things working

Bad

- Attack the PlayerBot
- Turn other TaskBots
- Interrupt normal work

0:45



TaskBot Intelligence

What's my motivation?

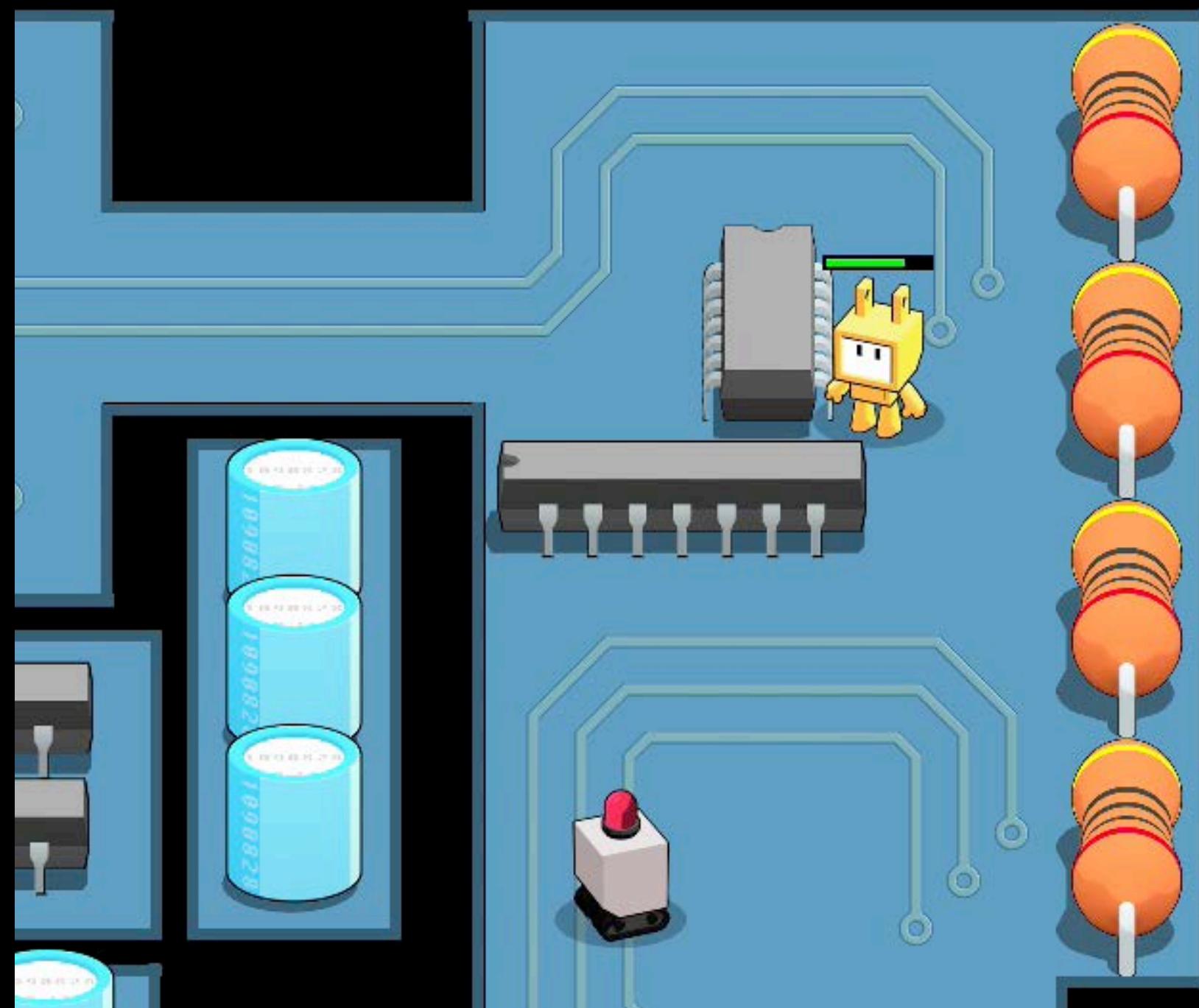
Good

- Keep things working

Bad

- Attack the PlayerBot
- Turn other TaskBots
- Interrupt normal work

0:45



TaskBot Intelligence

Determining a mandate

TaskBot Intelligence

Determining a mandate

Fuzzy Logic via **GKRuleSystem**

- Characters are engaged
- Simpler to maintain
- Emergent behavior

Fuzzy Rules

I'm a little fuzzy

Fuzzy Rules

I'm a little fuzzy

Model natural concepts

- Player is near
- Percentage of bad TaskBots is high

Fuzzy Rules

I'm a little fuzzy

Model natural concepts

- Player is near
- Percentage of bad TaskBots is high

Implemented via **FuzzyTaskBotRule**

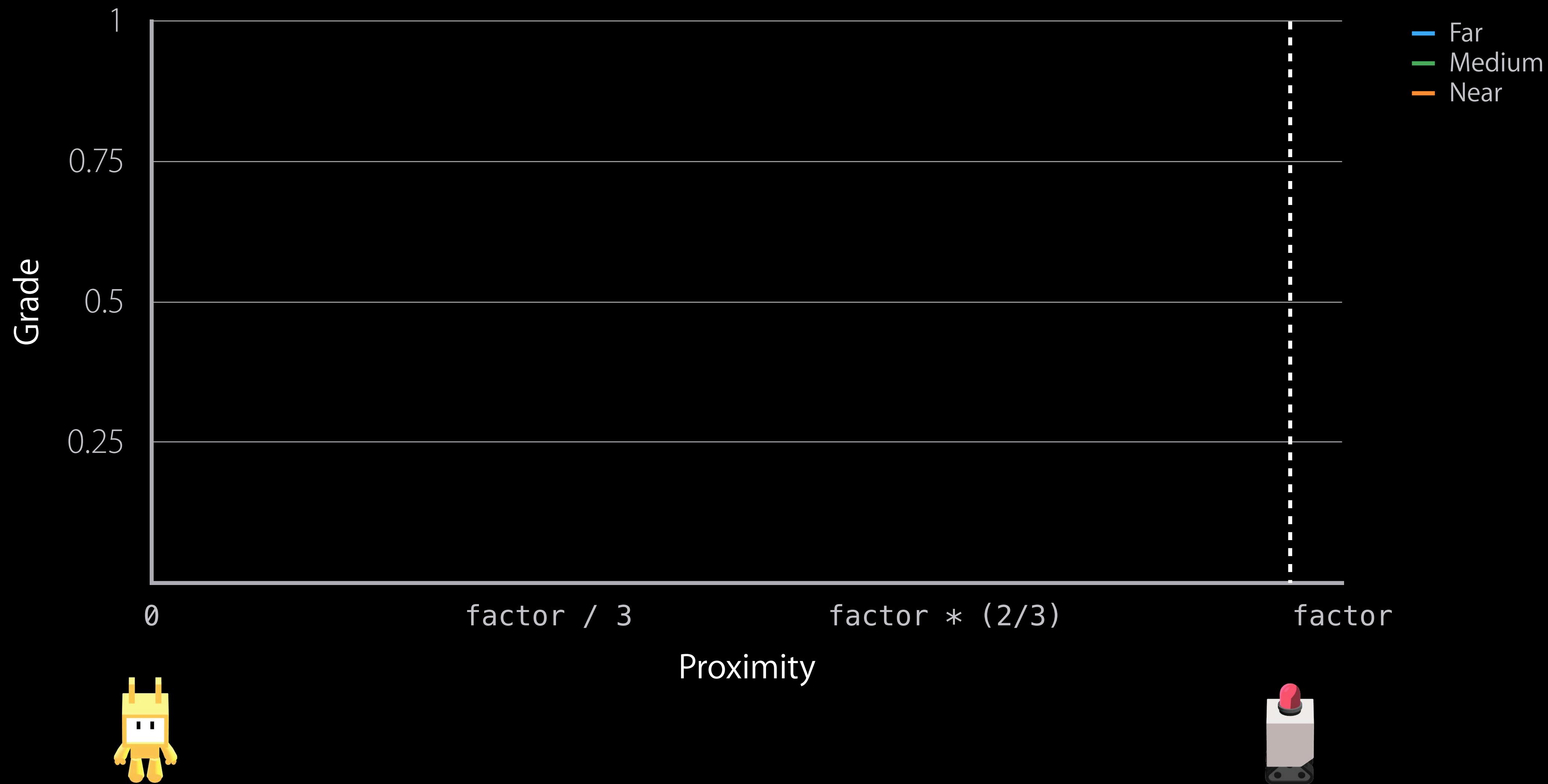
- Subclass of **GKRule**
- **fact** is asserted if **grade > 0**
- **grade** is a function of input

Fuzzy Rules

Functions for proximity rules

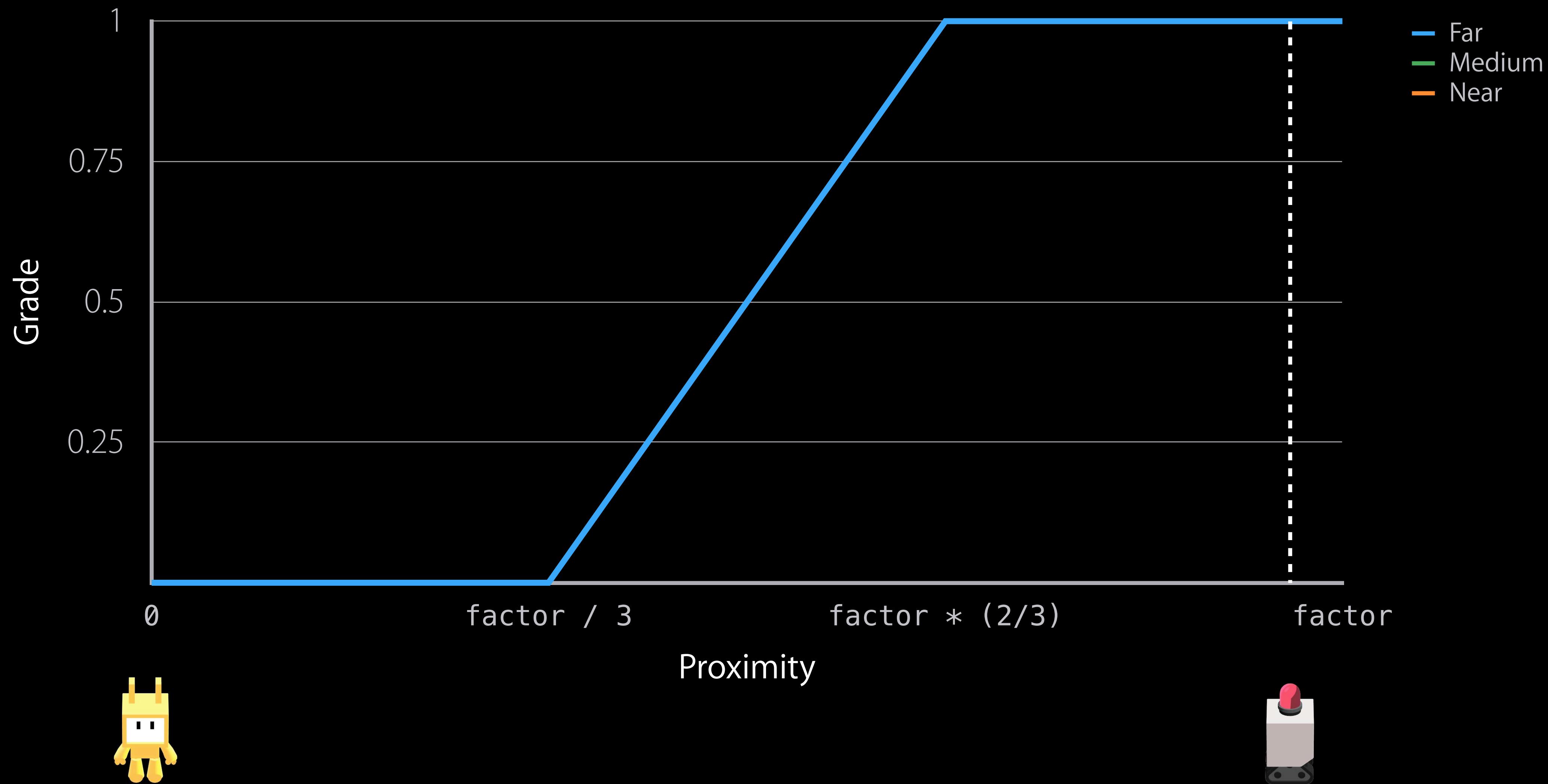
Fuzzy Rules

Functions for proximity rules



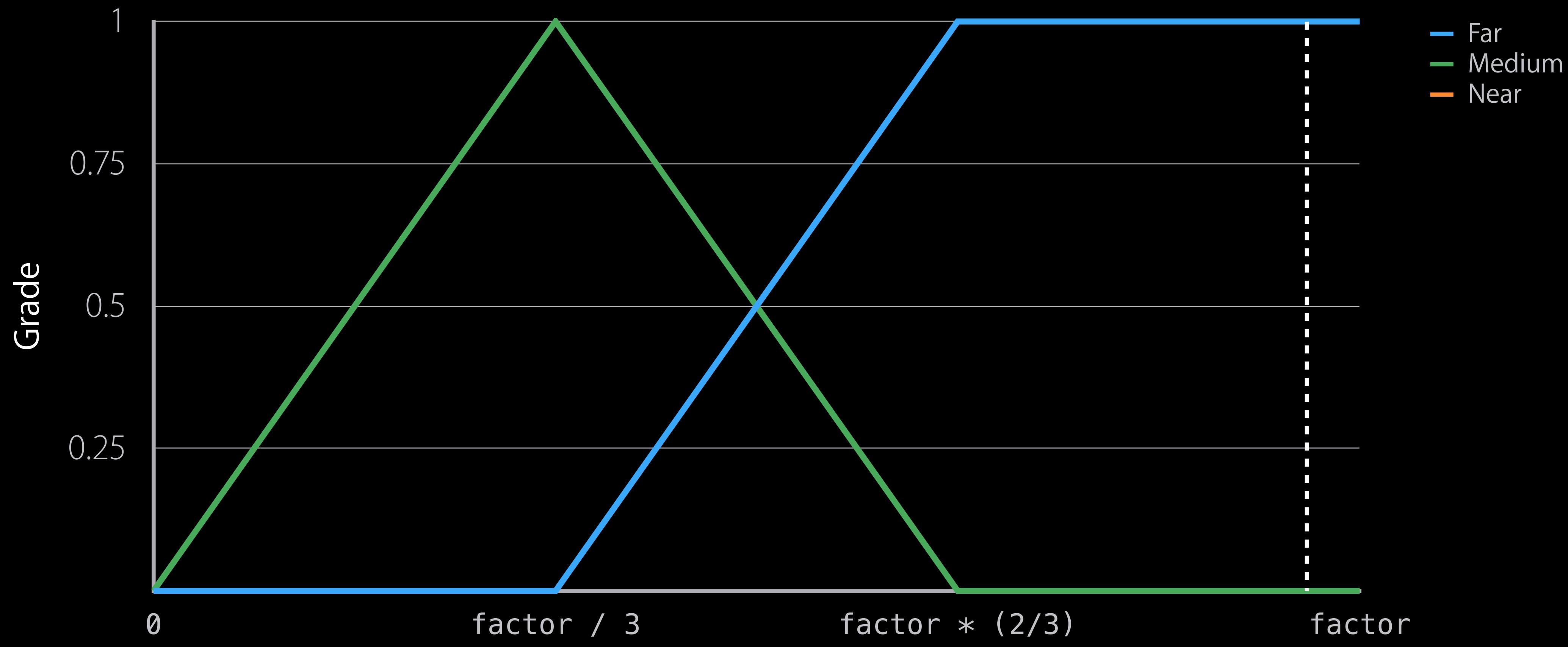
Fuzzy Rules

Functions for proximity rules



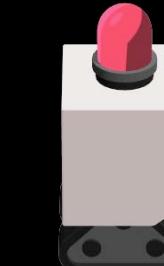
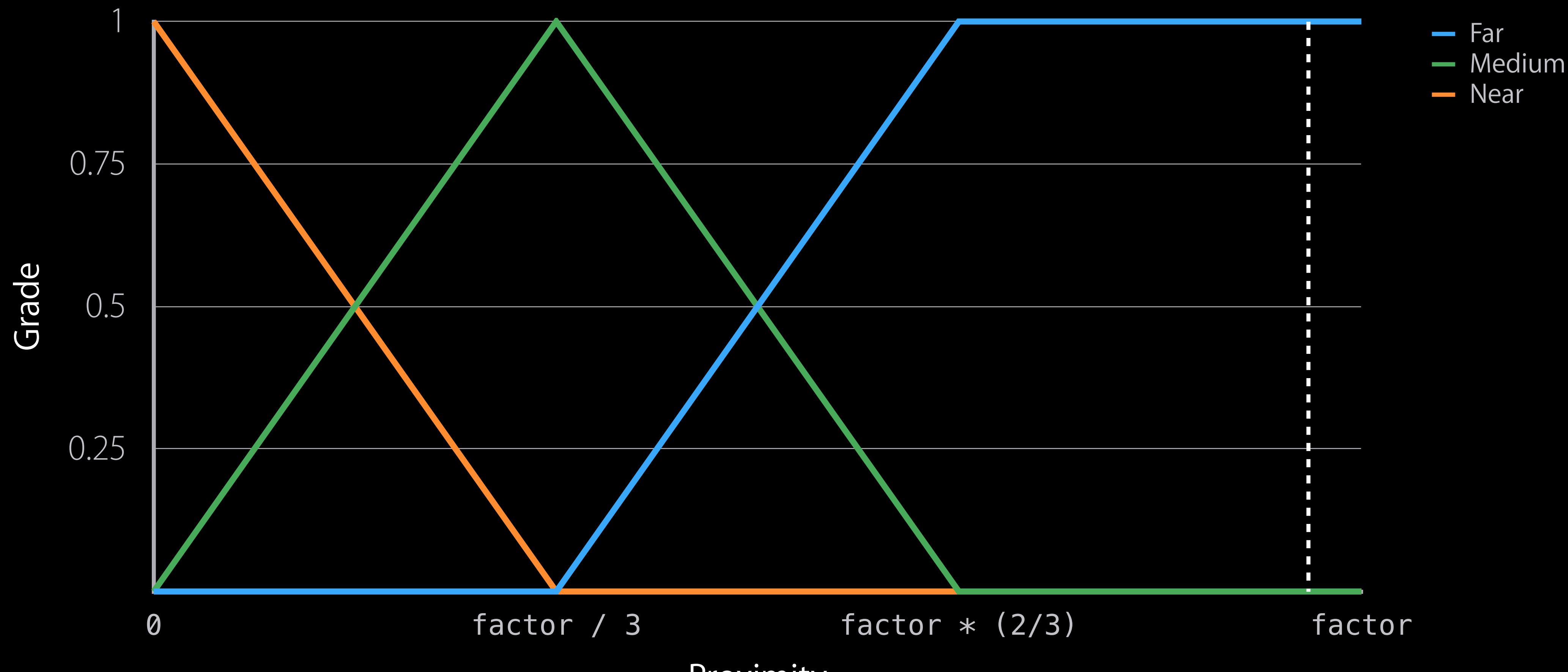
Fuzzy Rules

Functions for proximity rules



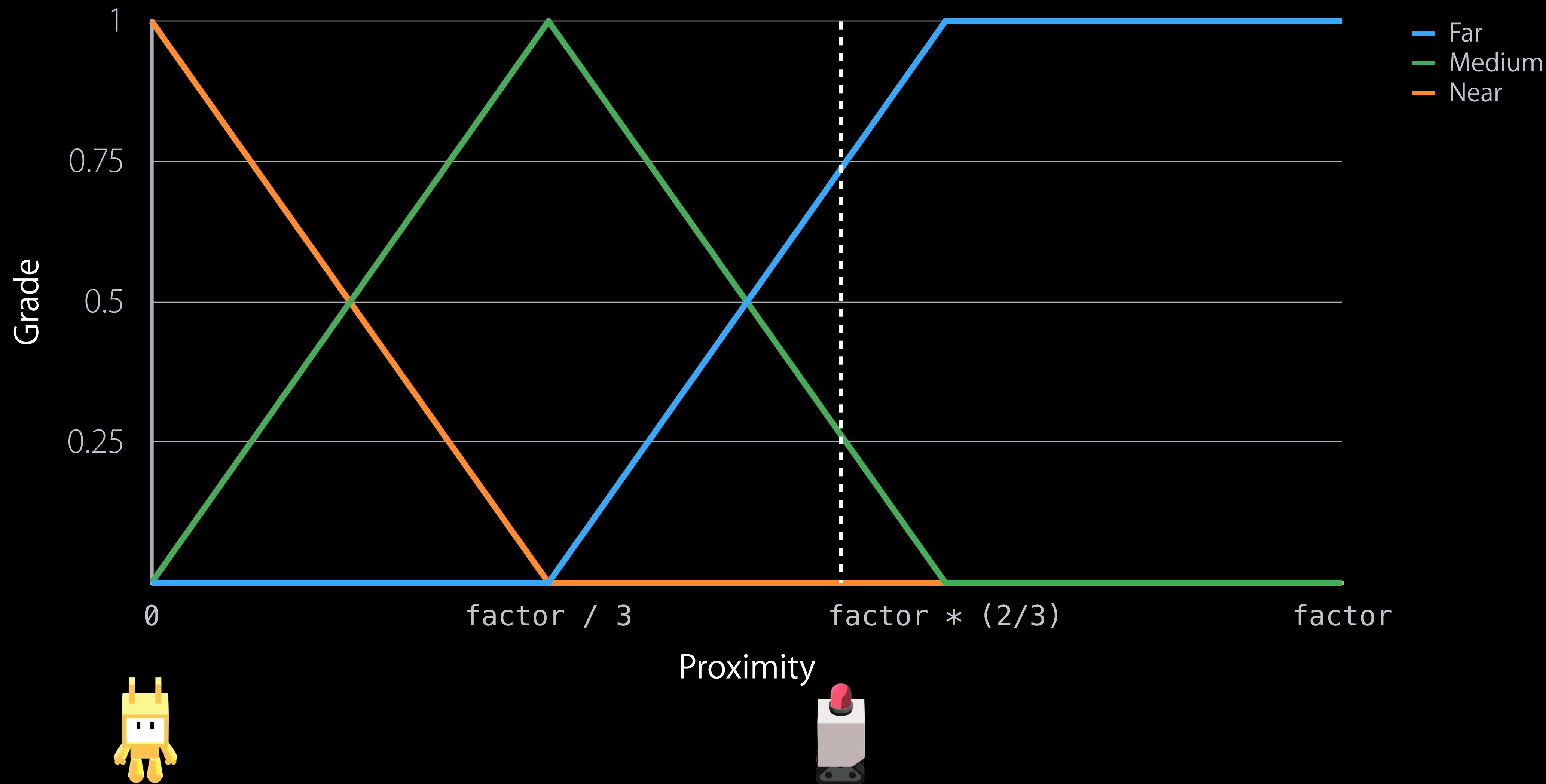
Fuzzy Rules

Functions for proximity rules



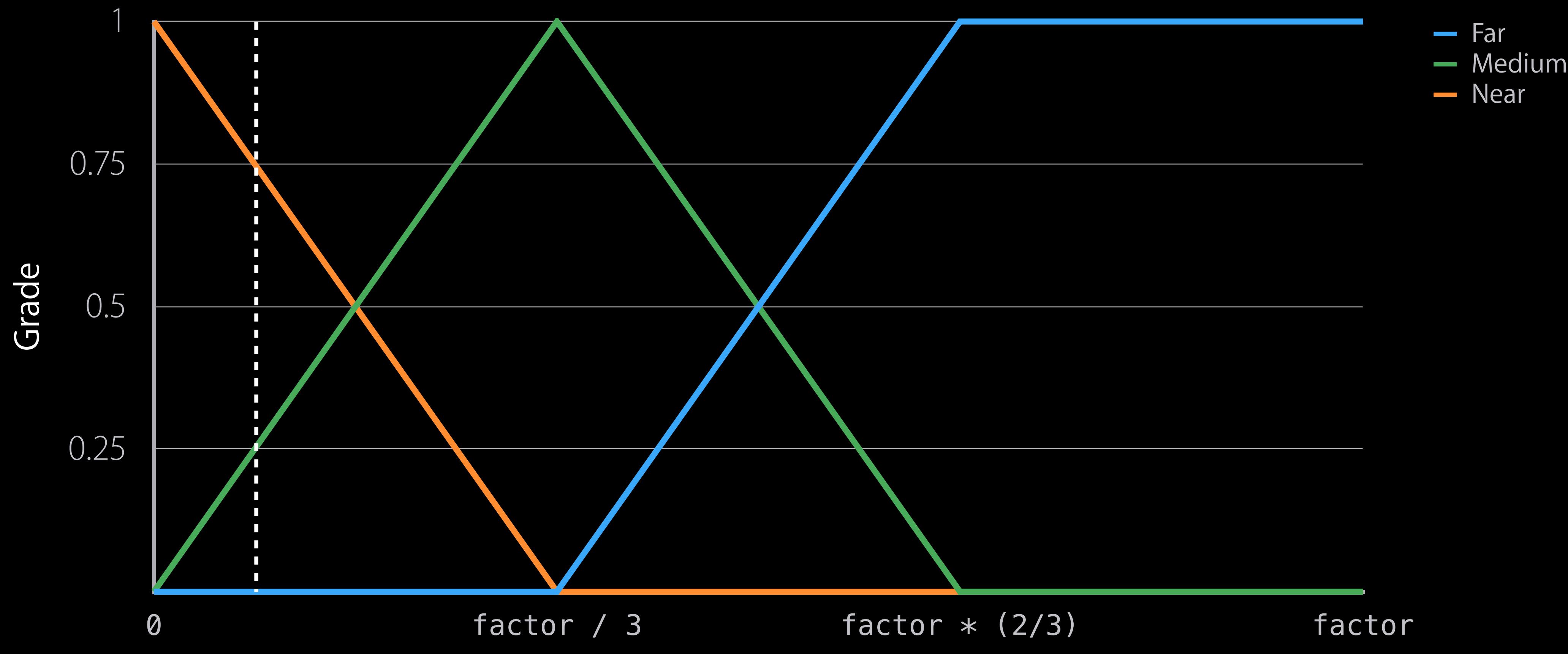
Fuzzy Rules

Functions for proximity rules



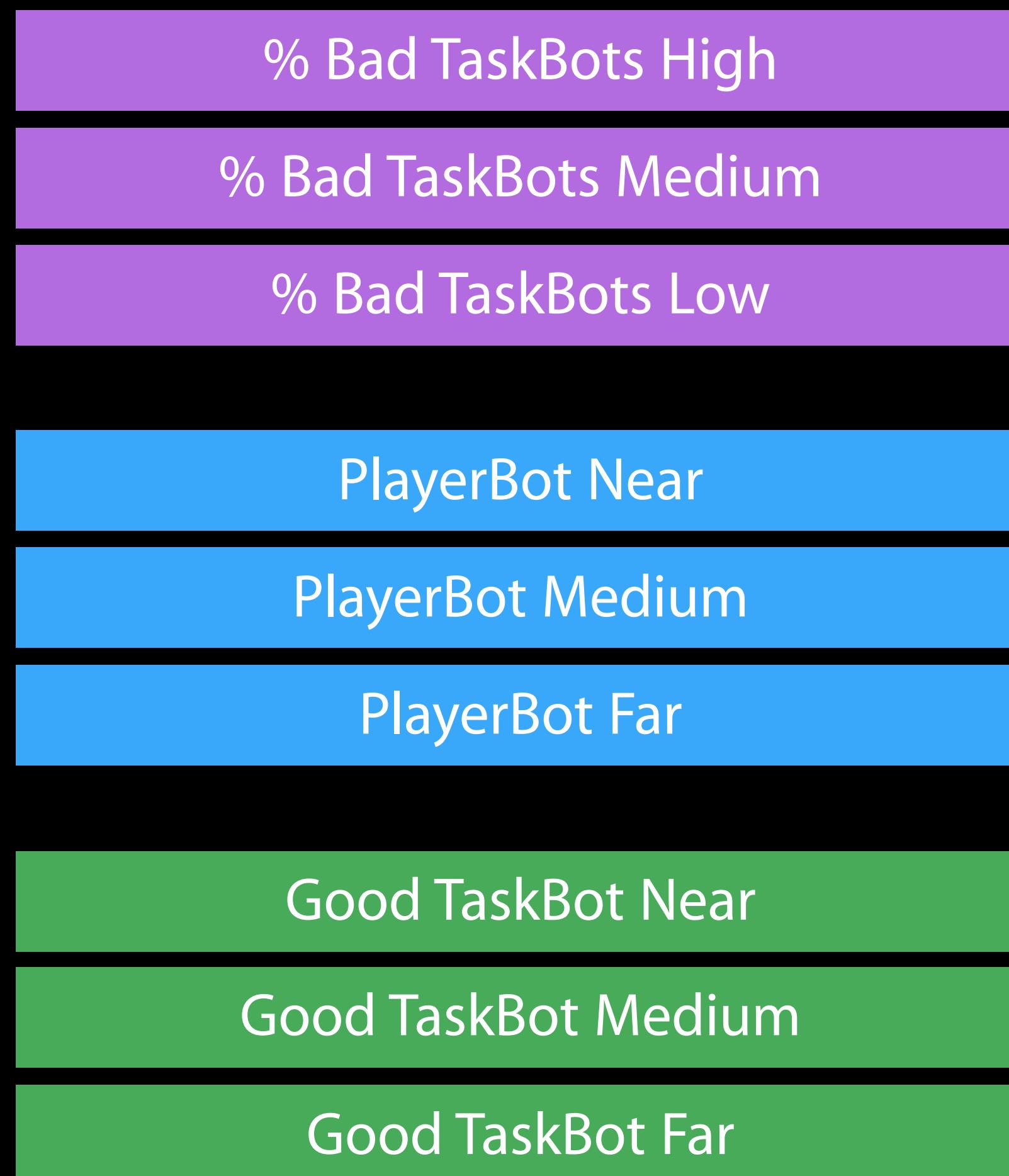
Fuzzy Rules

Functions for proximity rules



Fuzzy Rules

Making decisions



Fuzzy Rules

Making decisions

Evaluate the rules

% Bad TaskBots High	.60
% Bad TaskBots Medium	.40
% Bad TaskBots Low	.00
PlayerBot Near	.30
PlayerBot Medium	.70
PlayerBot Far	.00
Good TaskBot Near	.70
Good TaskBot Medium	.30
Good TaskBot Far	.00

Fuzzy Rules

Making decisions

Evaluate the rules

Combine facts

% Bad TaskBots High	.60
PlayerBot Medium	.70
Good TaskBot Medium	.30

Fuzzy Rules

Making decisions

Evaluate the rules

Combine facts

- `ruleSystem.minimumGradeForFacts(_:)`

Fuzzy Rules

Making decisions

Evaluate the rules

Combine facts

- `ruleSystem.minimumGradeForFacts(_:)`

Fuzzy Rules

Making decisions

Evaluate the rules

Hunt PlayerBot .30

Combine facts

Hunt PlayerBot .45

- `ruleSystem.minimumGradeForFacts(_:)`

Hunt TaskBot .00

Hunt TaskBot .20

Fuzzy Rules

Making decisions

Evaluate the rules

Combine facts

- `ruleSystem.minimumGradeForFacts(_:)`

Defuzzify rules



Fuzzy Rules

Making decisions

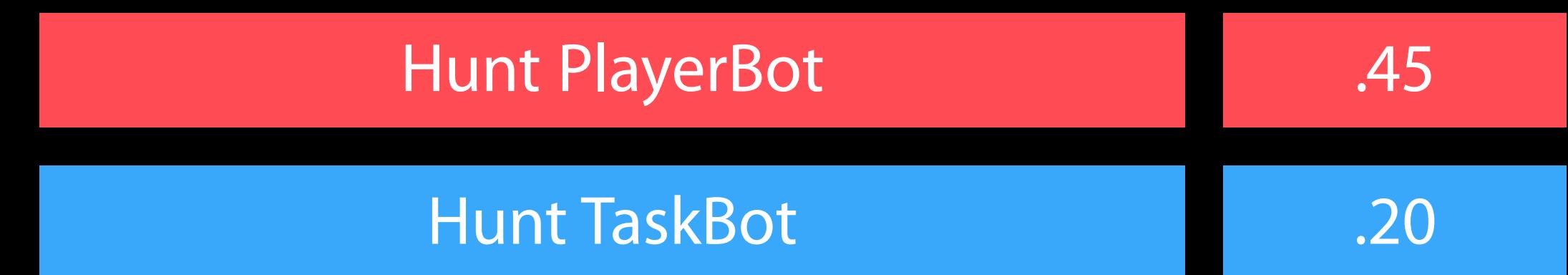
Evaluate the rules

Combine facts

- `ruleSystem.minimumGradeForFacts(_:)`

Defuzzify rules

- `facts.reduce(0.0, combine: max)`



Fuzzy Rules

Making decisions

Evaluate the rules

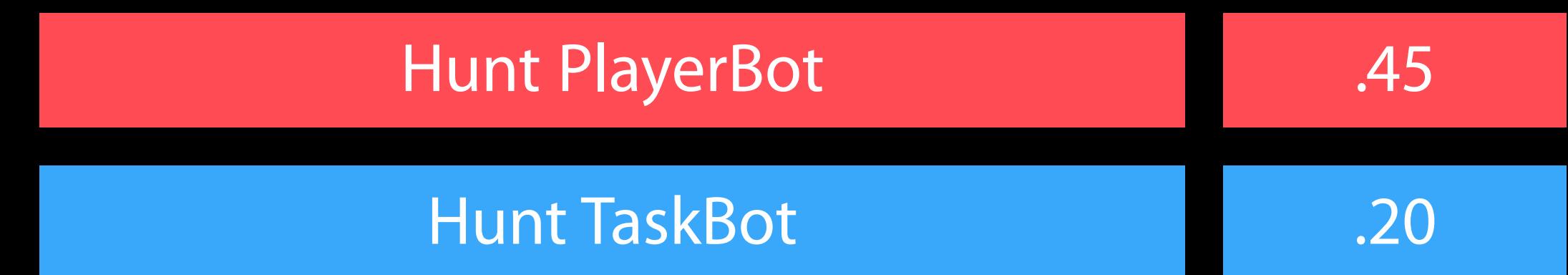
Combine facts

- `ruleSystem.minimumGradeForFacts(_:)`

Defuzzify rules

- `facts.reduce(0.0, combine: max)`

Hunt the PlayerBot



TaskBot Intelligence

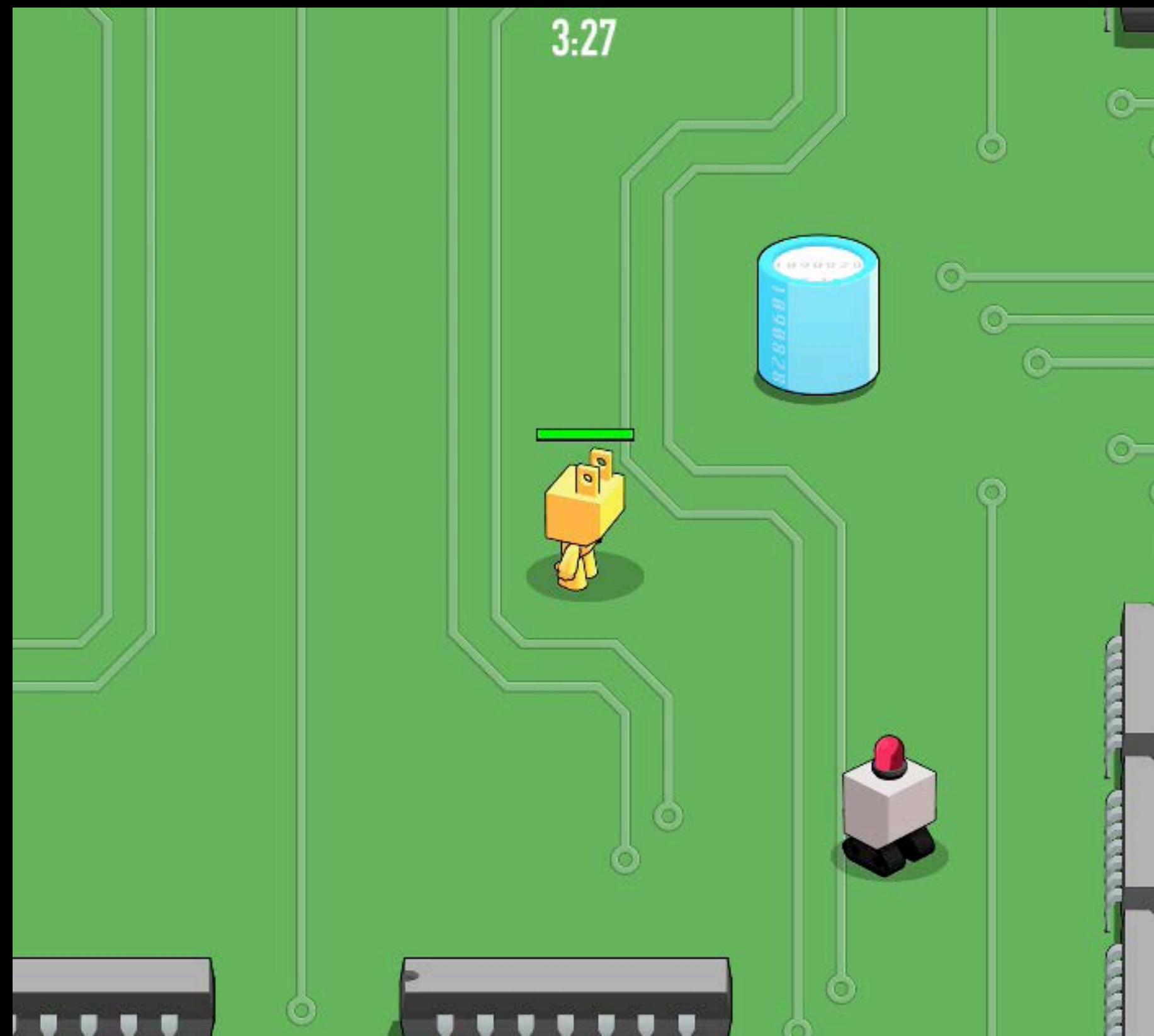
OK, hunt the PlayerBot

TaskBot Intelligence

OK, hunt the PlayerBot

Get to the **PlayerBot**

- Often straightforward

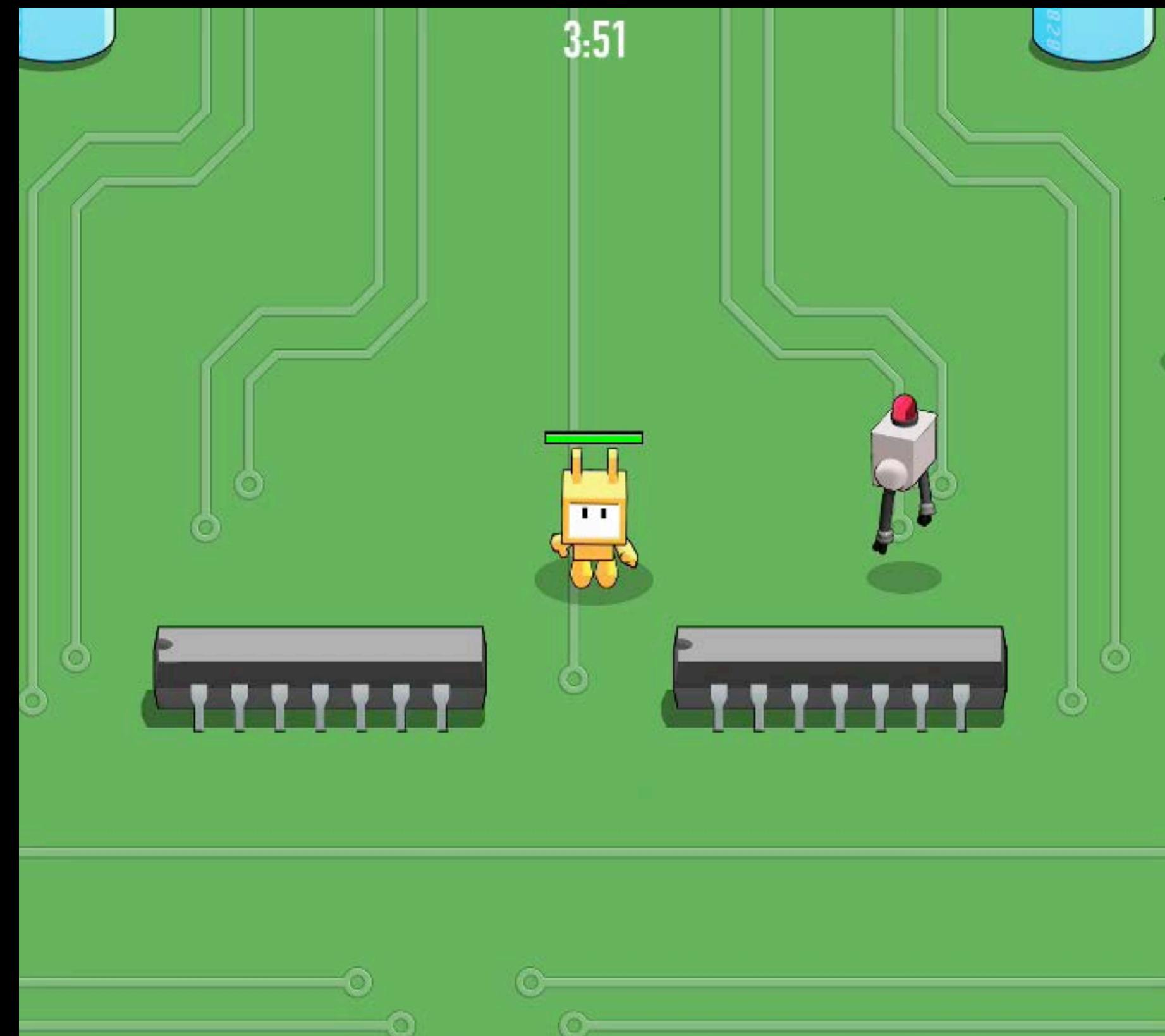


TaskBot Intelligence

OK, hunt the PlayerBot

Get to the **PlayerBot**

- Often straightforward
- Obstacles present a challenge

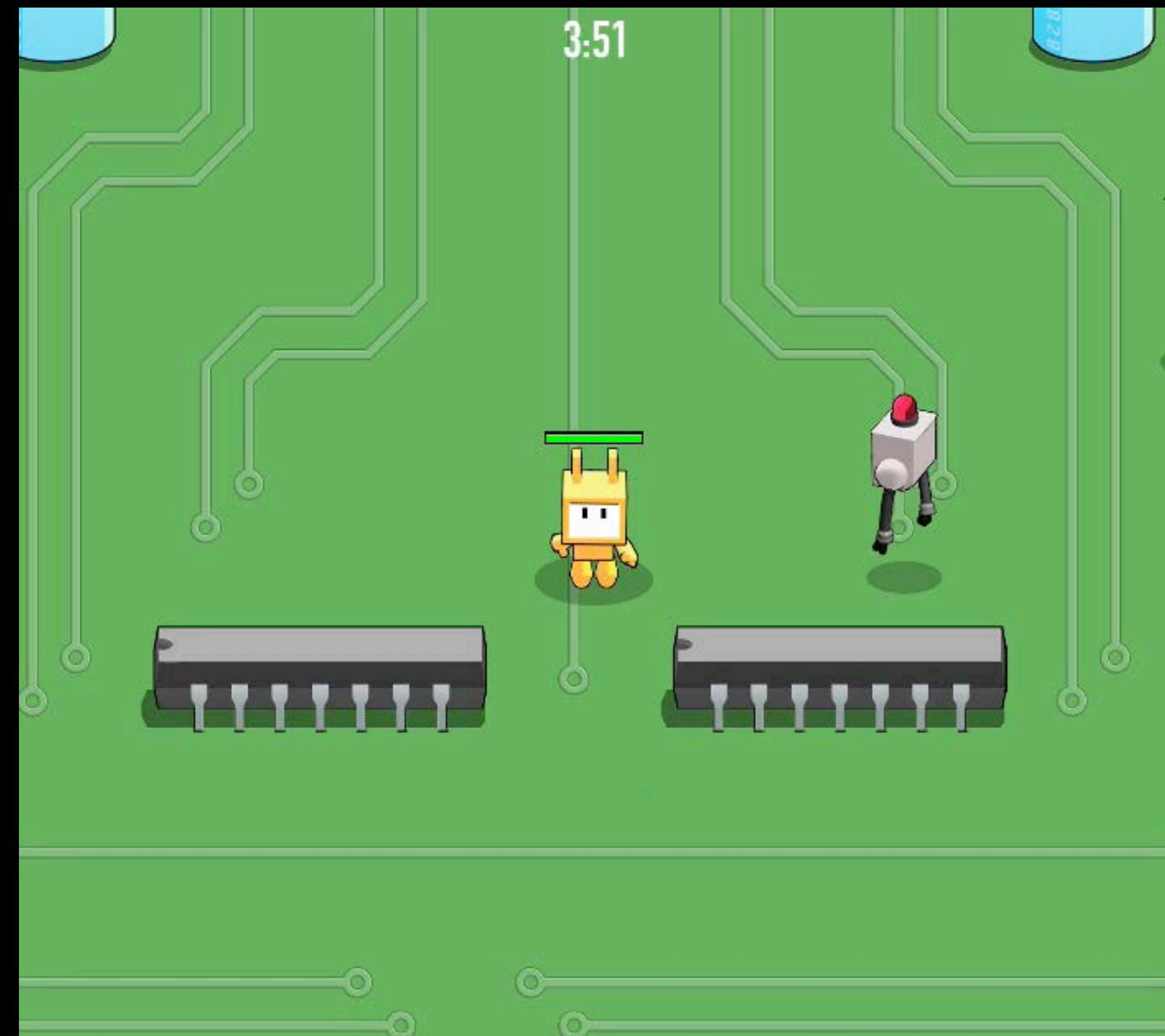


TaskBot Intelligence

OK, hunt the PlayerBot

Get to the **PlayerBot**

- Often straightforward
- Obstacles present a challenge



TaskBot Intelligence

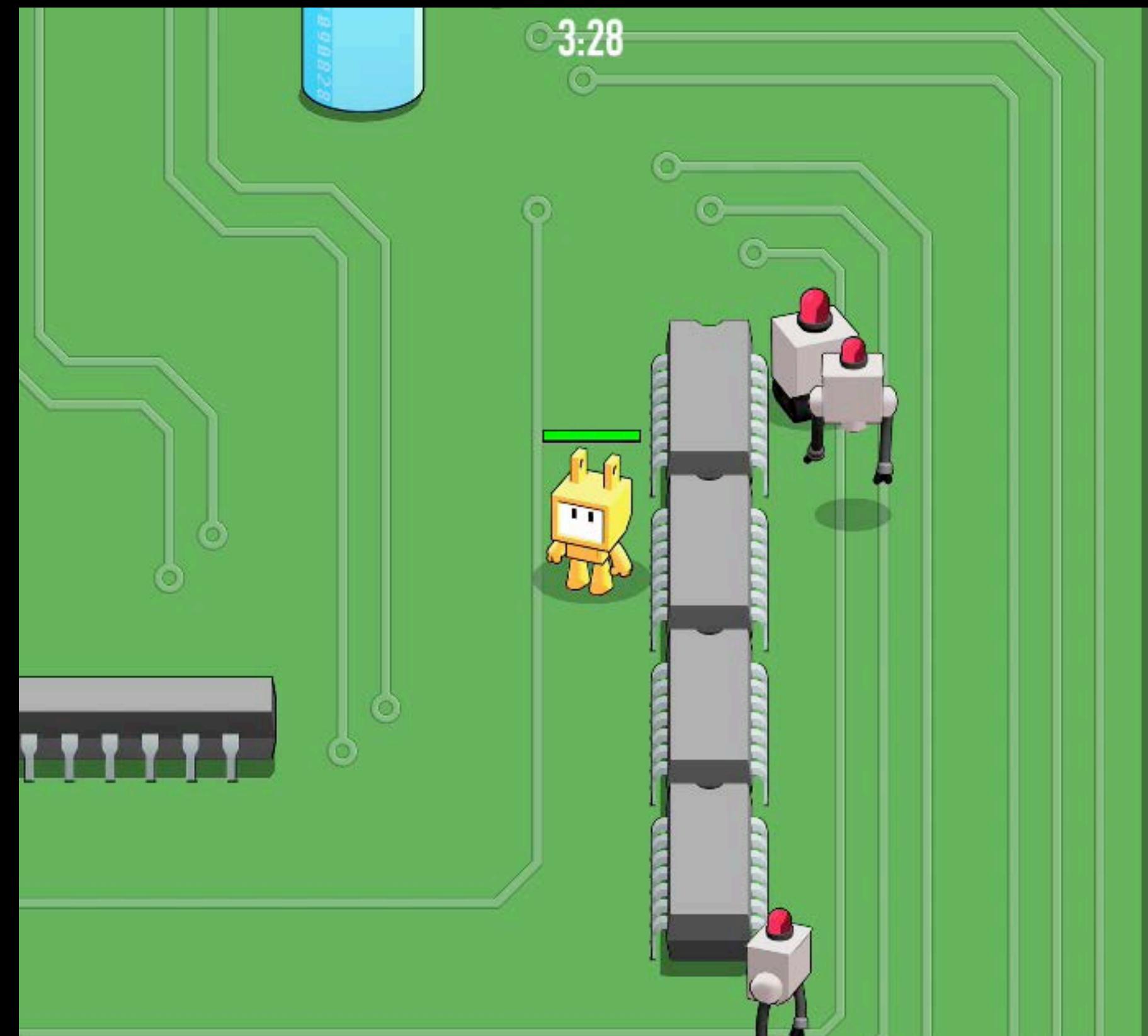
OK, hunt the PlayerBot

Get to the **PlayerBot**

- Often straightforward
- Obstacles present a challenge

Pathfinding via **GameplayKit**

- Conveniences for **SpriteKit**
- Up and running quickly



TaskBot Intelligence

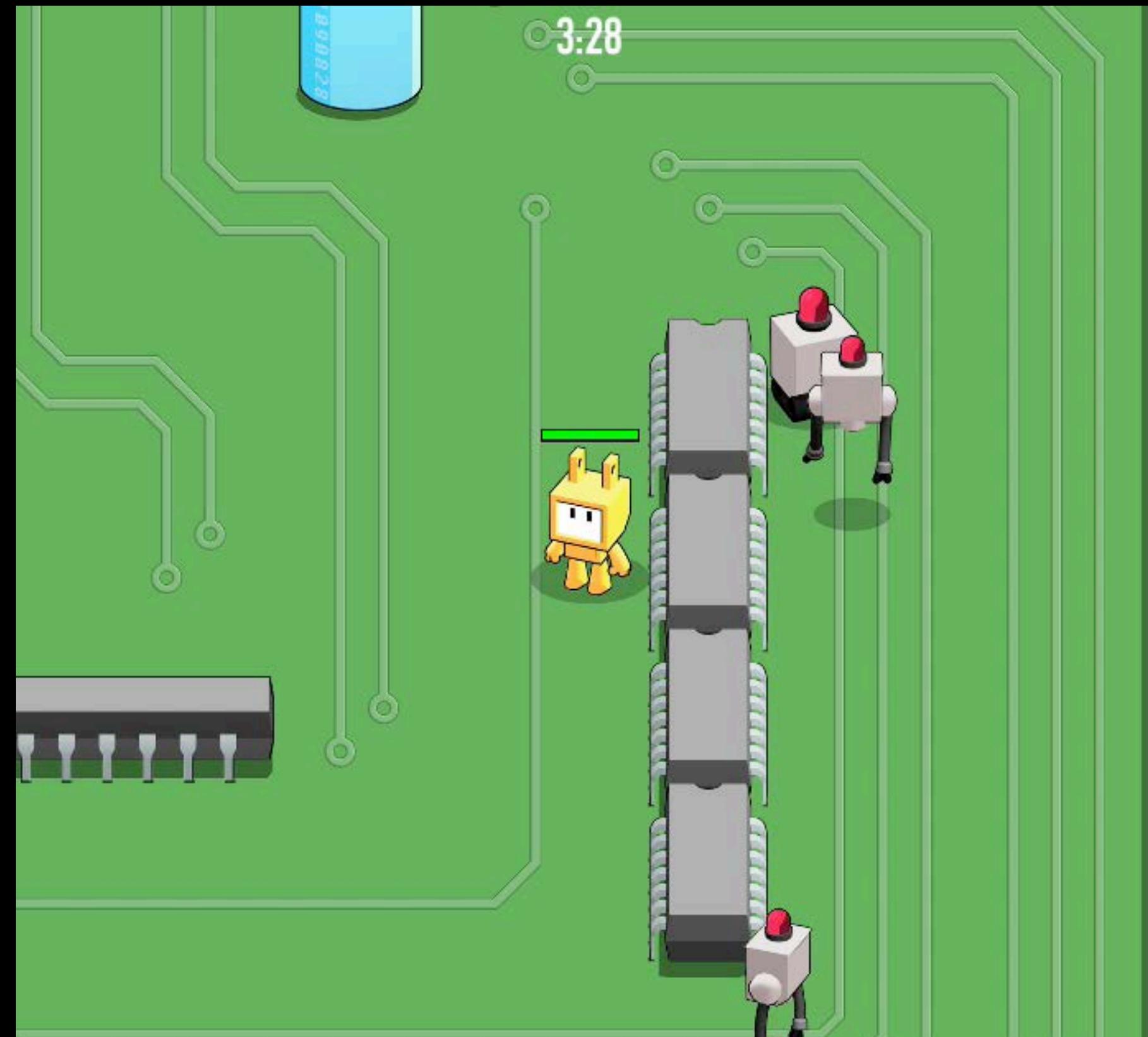
OK, hunt the PlayerBot

Get to the **PlayerBot**

- Often straightforward
- Obstacles present a challenge

Pathfinding via **GameplayKit**

- Conveniences for **SpriteKit**
- Up and running quickly



Pathfinding

A way around obstacles

Pathfinding

A way around obstacles

```
// Convert node physics bodies into GKPolygonObstacles.  
let obstacles = SKNode.obstaclesFromNodePhysicsBodies(self["//obstacles"] )
```

Pathfinding

A way around obstacles

```
// Convert node physics bodies into GKPolygonObstacles.  
let obstacles = SKNode.obstaclesFromNodePhysicsBodies(self["//obstacles"] )  
  
// Construct a GKObstacleGraph.  
let graph = GKObstacleGraph(obstacles: obstacles, bufferRadius: 30.0)
```

Pathfinding

A way around obstacles

```
// Convert node physics bodies into GKPolygonObstacles.  
let obstacles = SKNode.obstaclesFromNodePhysicsBodies(self["//obstacles"] )  
  
// Construct a GKObstacleGraph.  
let graph = GKObstacleGraph(obstacles: obstacles, bufferRadius: 30.0)  
  
// Connect the TaskBot and PlayerBot via their positions.  
let startNode = graph.connectNodeUsingObstacles(taskBot.position)  
let endNode = graph.connectNodeUsingObstacles(playerBot.position)
```

Pathfinding

A way around obstacles

```
// Convert node physics bodies into GKPolygonObstacles.  
let obstacles = SKNode.obstaclesFromNodePhysicsBodies(self["//obstacles"] )  
  
// Construct a GKObstacleGraph.  
let graph = GKObstacleGraph(obstacles: obstacles, bufferRadius: 30.0)  
  
// Connect the TaskBot and PlayerBot via their positions.  
let startNode = graph.connectNodeUsingObstacles(taskBot.position)  
let endNode = graph.connectNodeUsingObstacles(playerBot.position)  
  
// Find the path.  
let pathNodes = graph.findPathFromNode(startNode,  
    toNode: endNode) as! [GKGraphNode2D]
```

TaskBot Intelligence

All mapped out with no way to go

TaskBot Intelligence

All mapped out with no way to go

GKAgent2D to the rescue

- Configure a **GKBehavior**
- **GKGoal** provides path goals
- SpriteKit updates via delegate

Agent Behavior

A smooth motivator

Agent Behavior

A smooth motivator

```
// Construct a path from the pathNodes.  
let path = GKPath(graphNodes: pathNodes, radius: 20.0)
```

Agent Behavior

A smooth motivator

```
// Construct a path from the pathNodes.  
let path = GKPath(graphNodes: pathNodes, radius: 20.0)  
  
// Create a new behavior.  
let behavior = GKBehavior()
```

Agent Behavior

A smooth motivator

```
// Construct a path from the pathNodes.  
let path = GKPath(graphNodes: pathNodes, radius: 20.0)  
  
// Create a new behavior.  
let behavior = GKBehavior()  
  
// Add goals to follow and stay on the path.  
let followPathGoal = GKGoal(toFollowPath: path, maxPredictionTime: 1.0,  
    forward: true)  
behavior.setWeight(1.0, forGoal: followPathGoal)  
  
let stayOnPathGoal = GKGoal(toStayOnPath: path, maxPredictionTime: 1.0)  
behavior.setWeight(1.0, forGoal: stayOnPathGoal)
```

Agent Behavior

Move along, please

Agent Behavior

Move along, please

```
// Assign the behavior to the agent.  
agent.behavior = behavior
```

Agent Behavior

Move along, please

```
// Assign the behavior to the agent.
```

```
agent.behavior = behavior
```

```
// Agent will update based on the behavior.
```

```
// Its delegate is notified before and after changes.
```

Agent Behavior

Move along, please

```
// Assign the behavior to the agent.  
agent.behavior = behavior  
  
// Agent will update based on the behavior.  
// Its delegate is notified before and after changes.  
  
func agentWillUpdate(agent: GKAgent) {  
    // TIP: Update agent position to match node position in SpriteKit.  
}
```

Agent Behavior

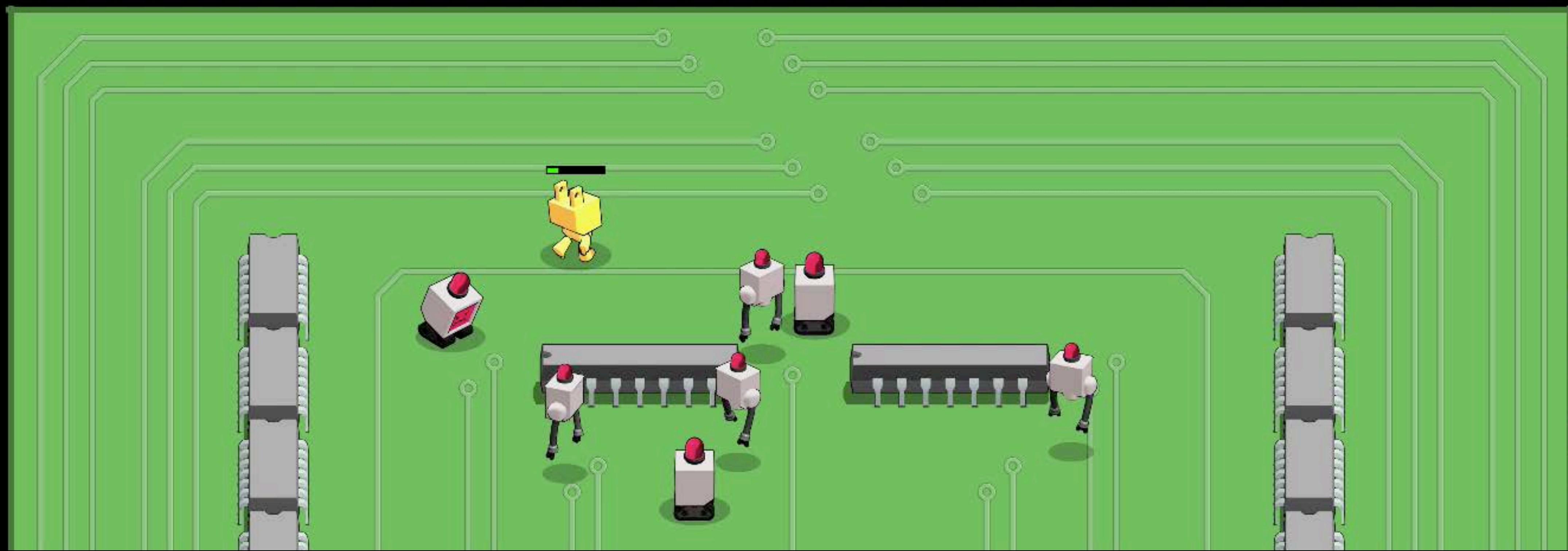
Move along, please

```
// Assign the behavior to the agent.  
agent.behavior = behavior  
  
// Agent will update based on the behavior.  
// Its delegate is notified before and after changes.  
  
func agentWillUpdate(agent: GKAgent) {  
    // TIP: Update agent position to match node position in SpriteKit.  
}  
  
func agentDidUpdate(agent: GKAgent) {  
    // TIP: Update node position in SpriteKit to match agent position.  
}
```

TaskBot Intelligence

Just the beginning

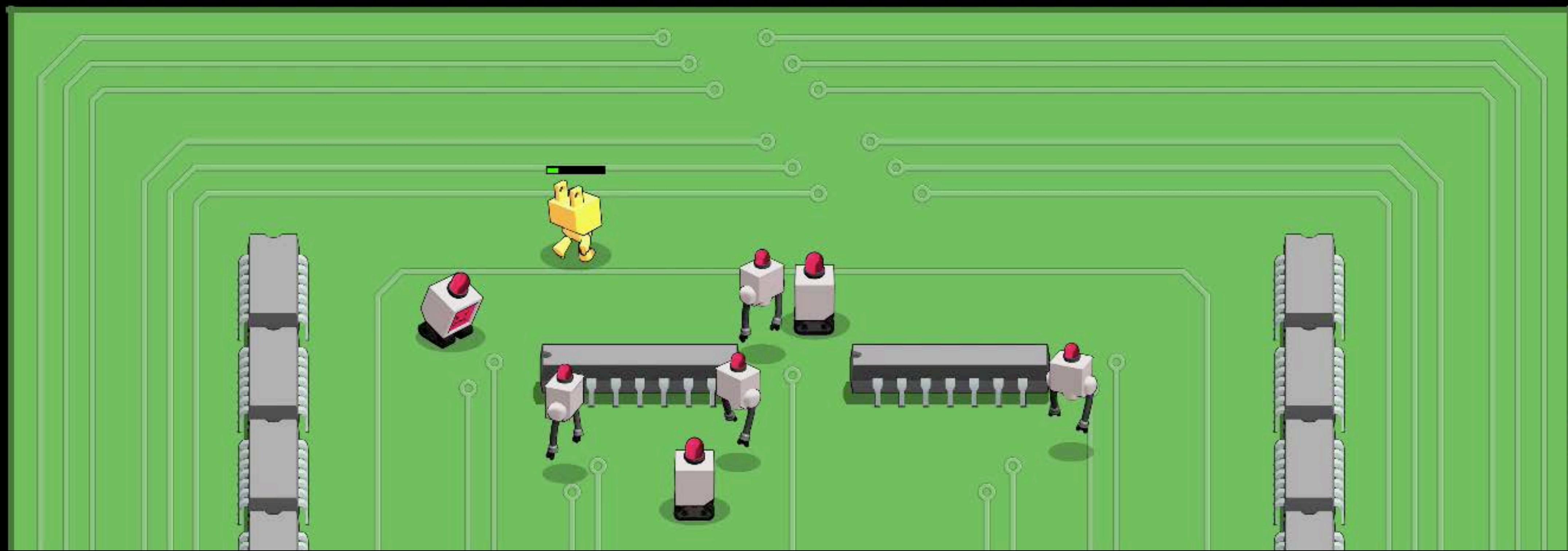
2:20



TaskBot Intelligence

Just the beginning

2:20



Deployment and Delivery

Michael DeWitt Sample Code Engineer

Time to Fun

Initial Download

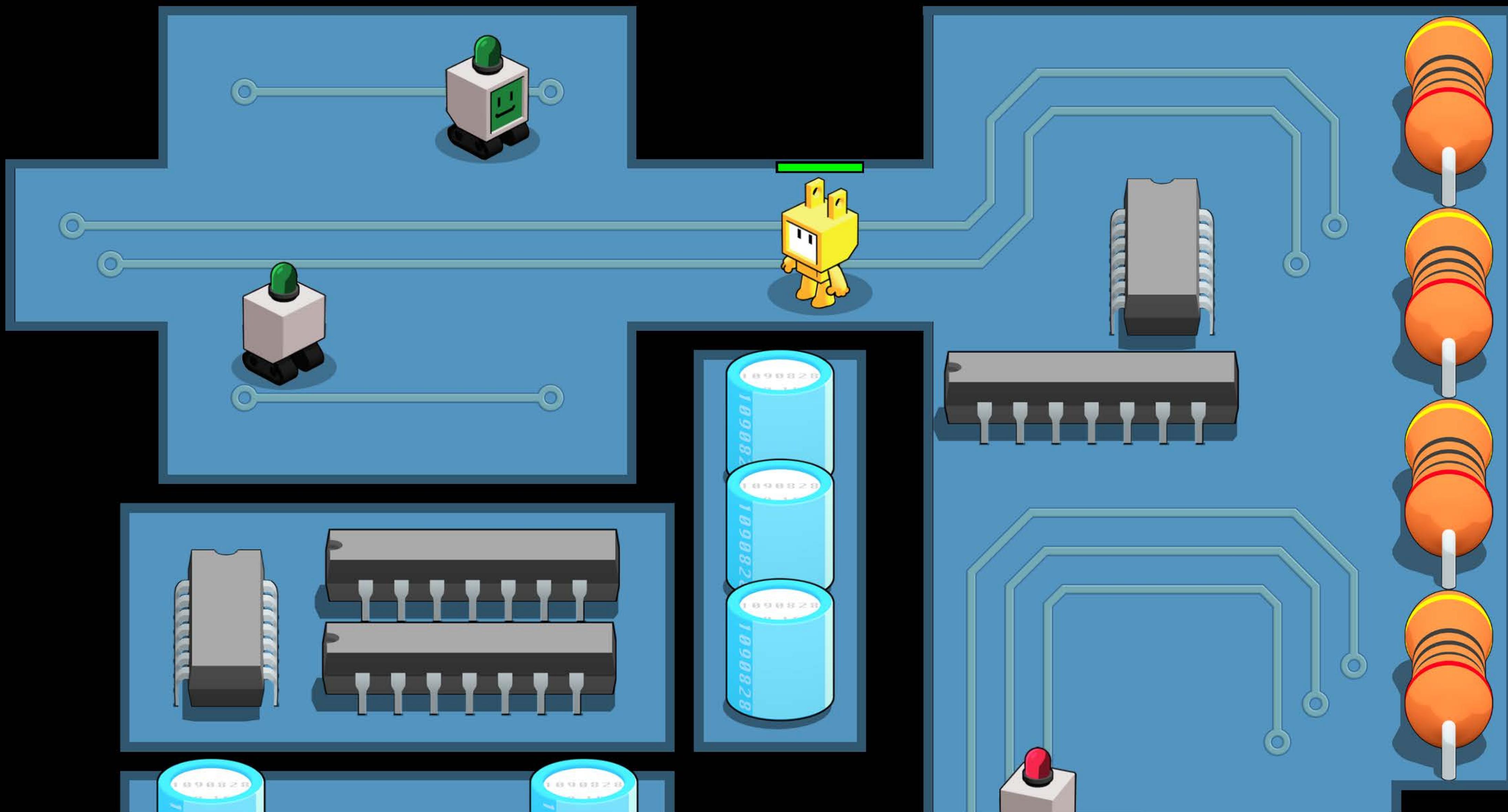
Initial Download

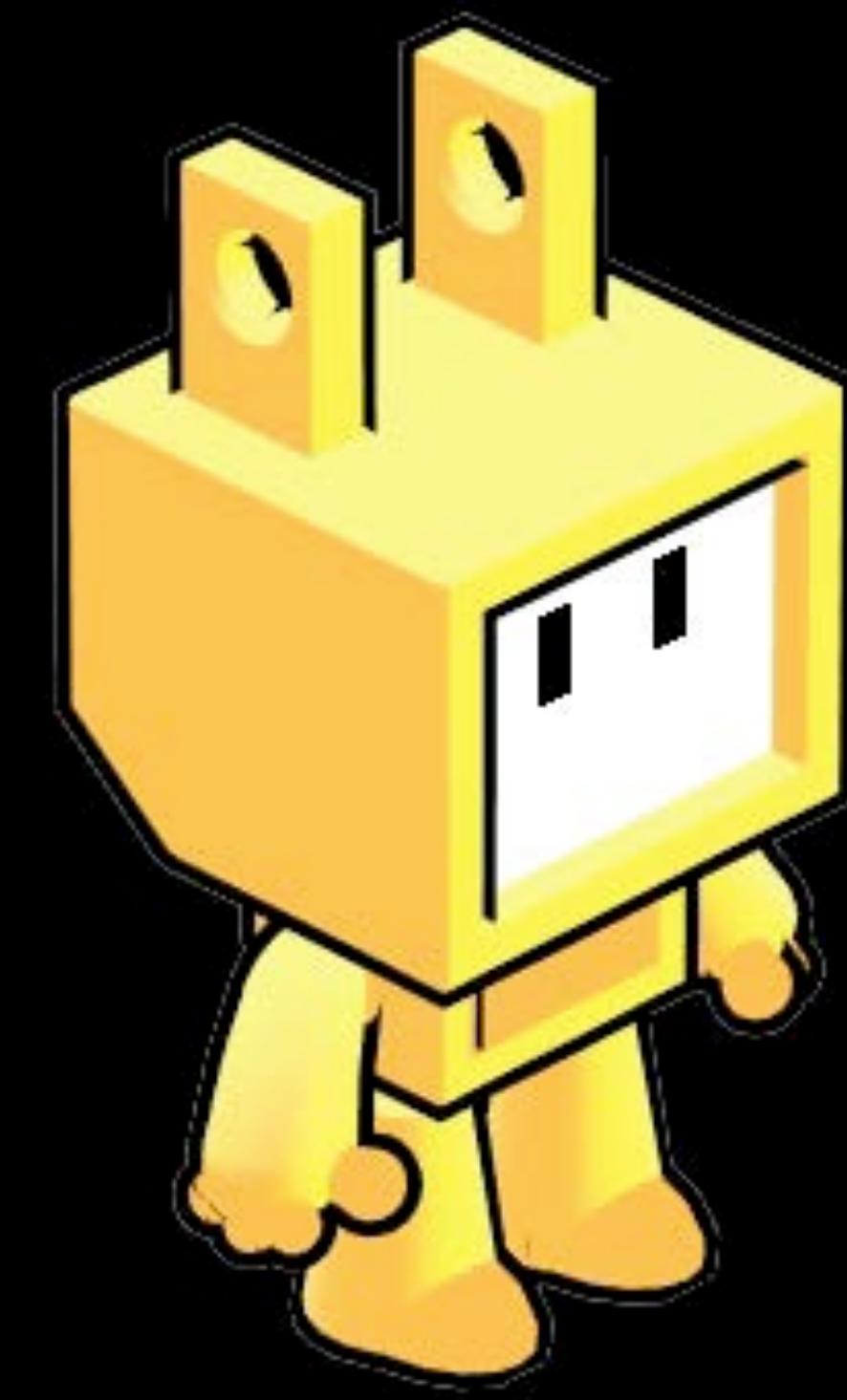
This item is over 100MB.
DemoBots will not download
until you connect to Wi-Fi.

[Cancel](#)

[OK](#)

2:43



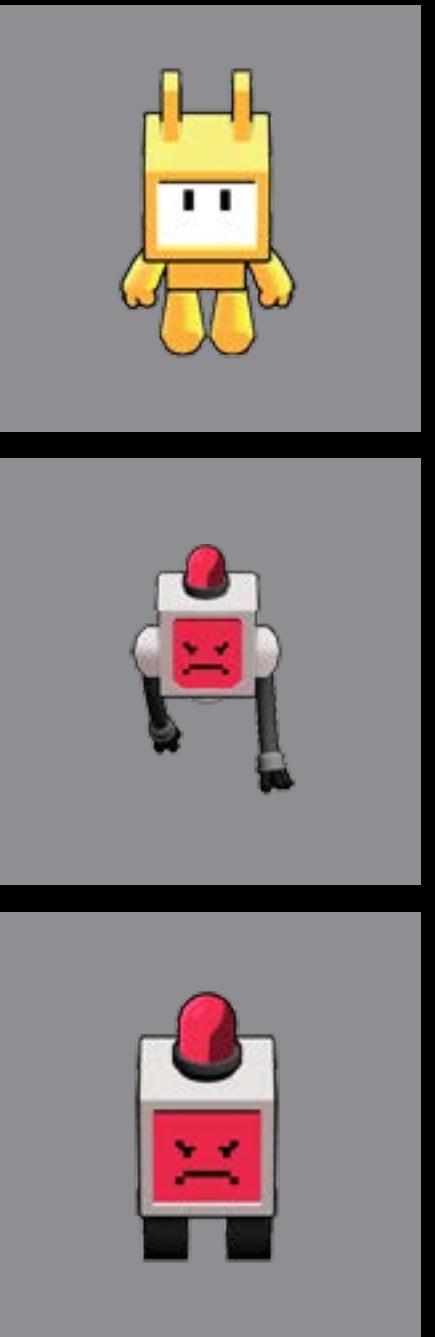




6 MB

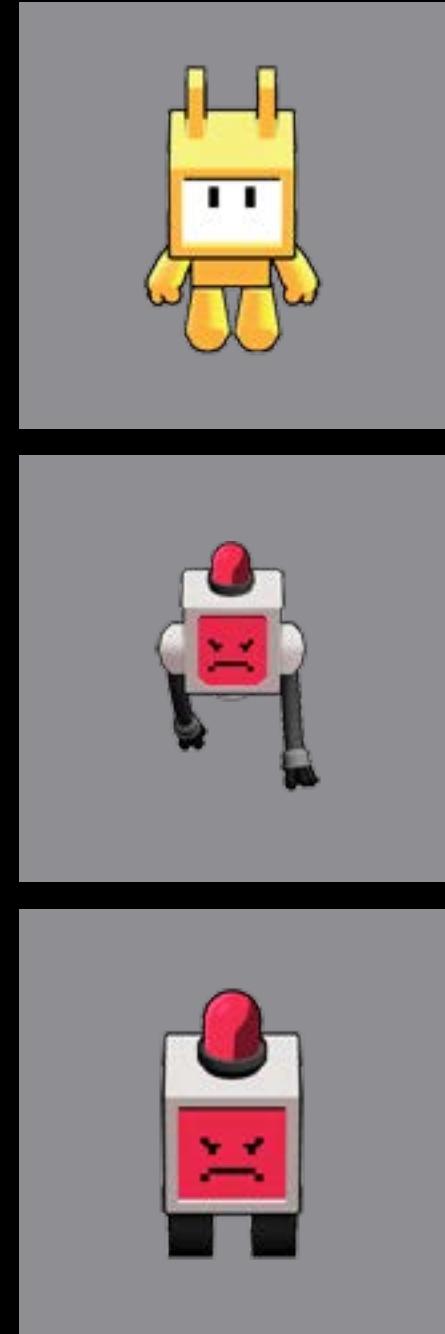
3 Bots

6 MB ×

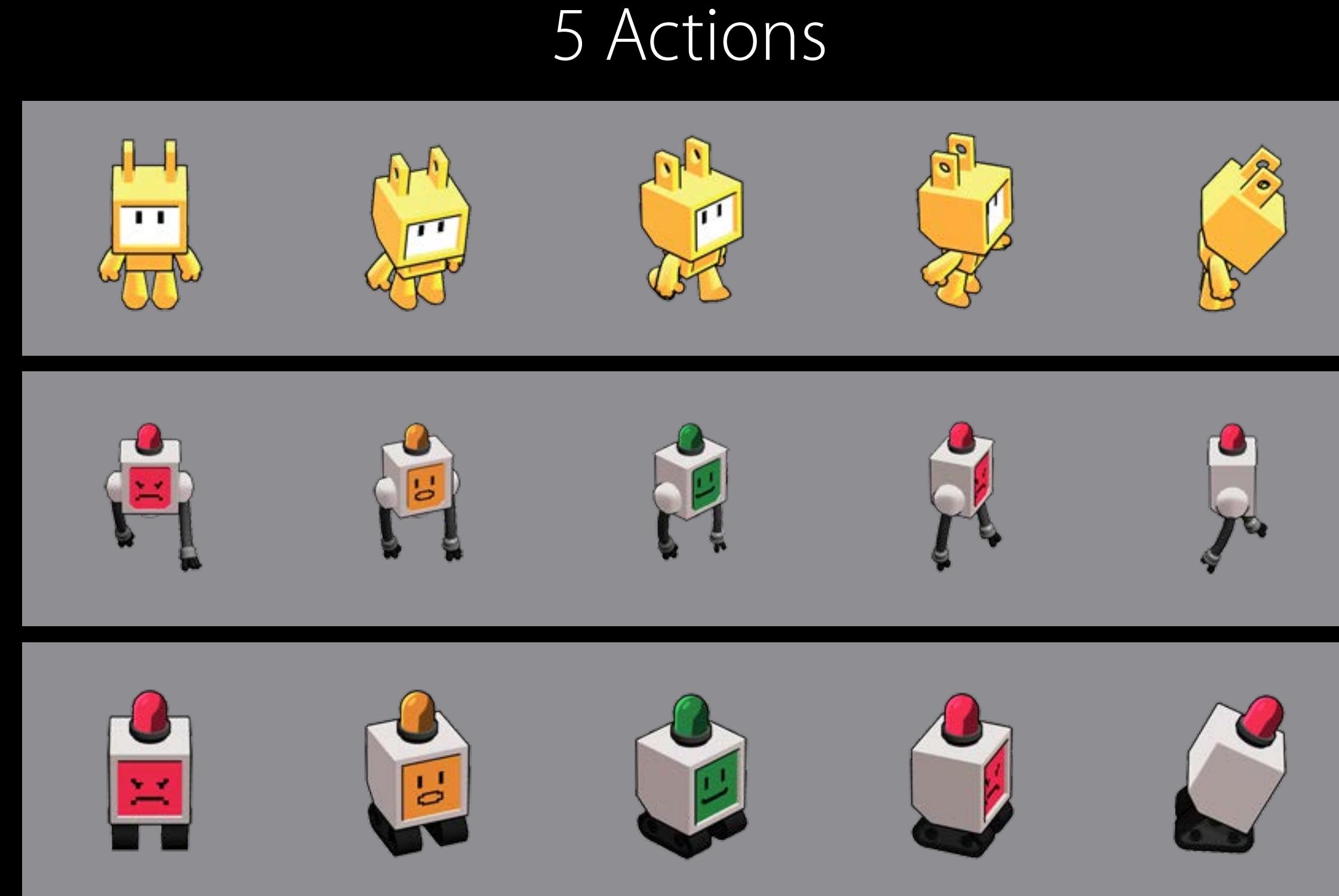


6 MB

3 Bots

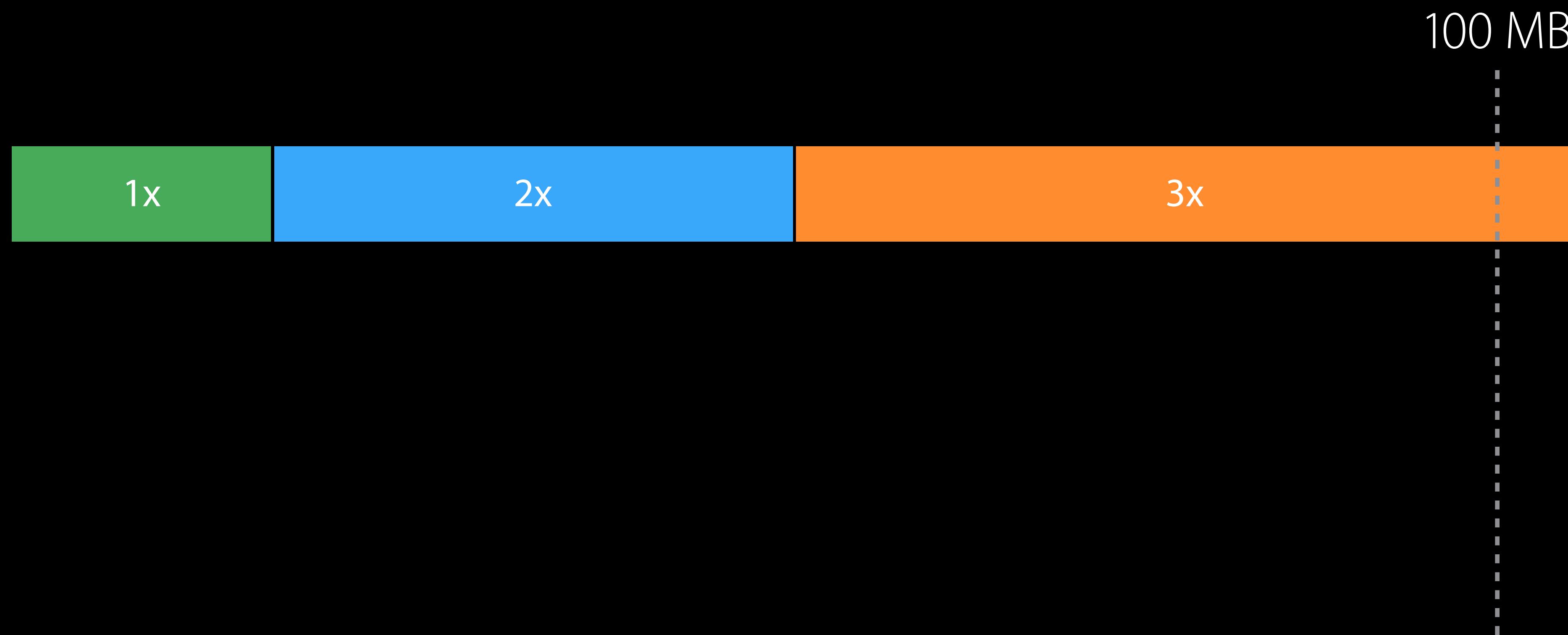


x

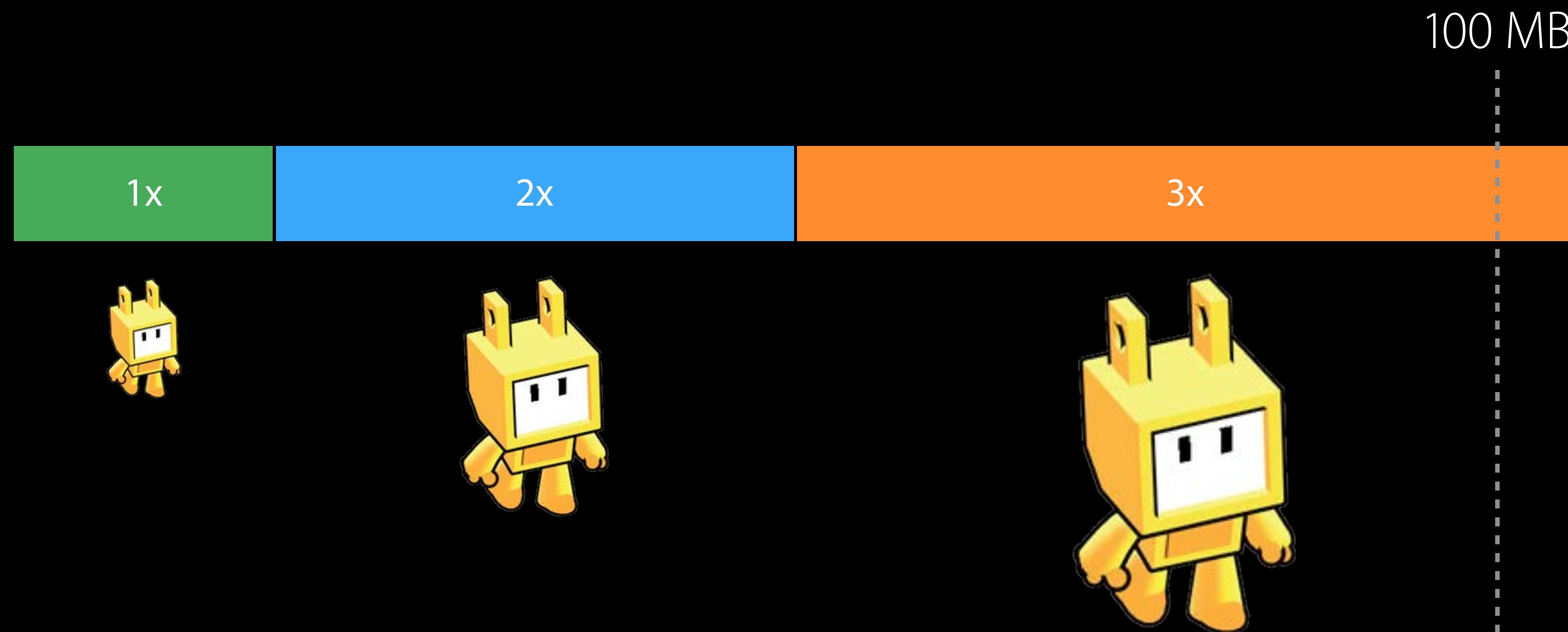


5 Actions

In Your App

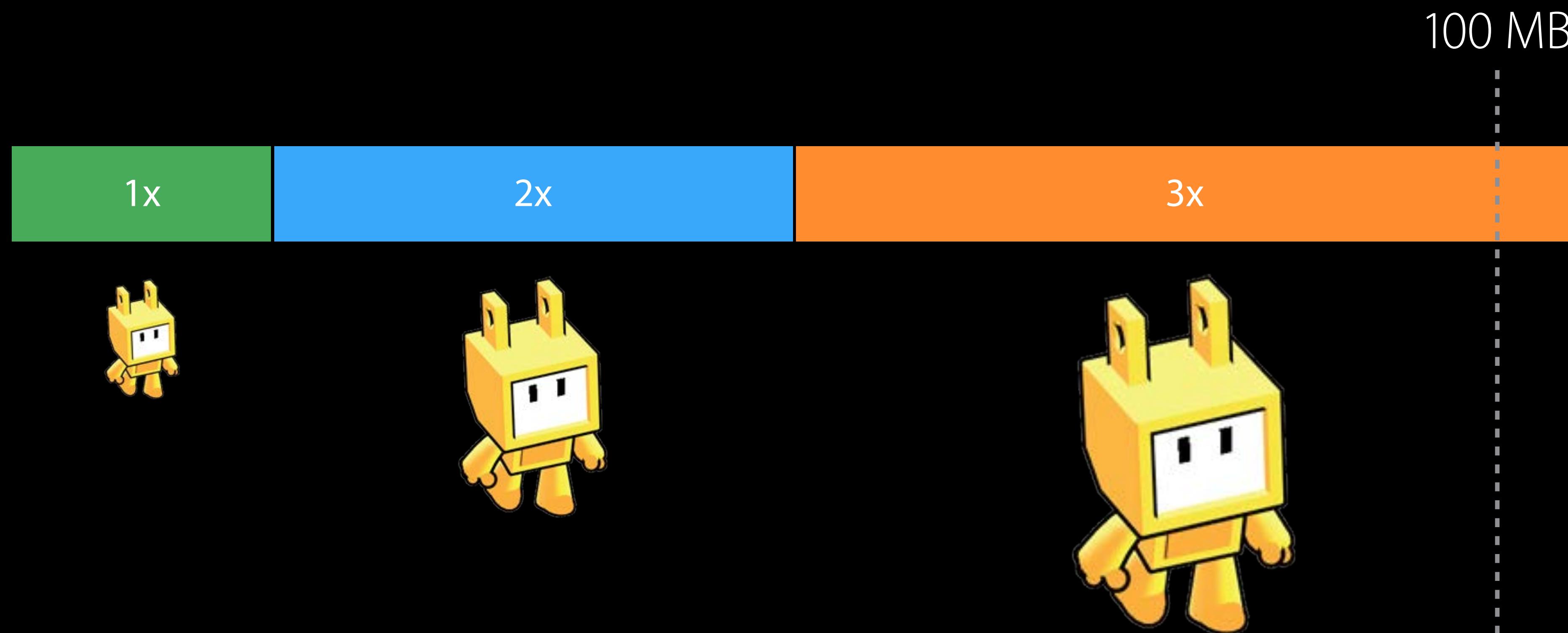


In Your App



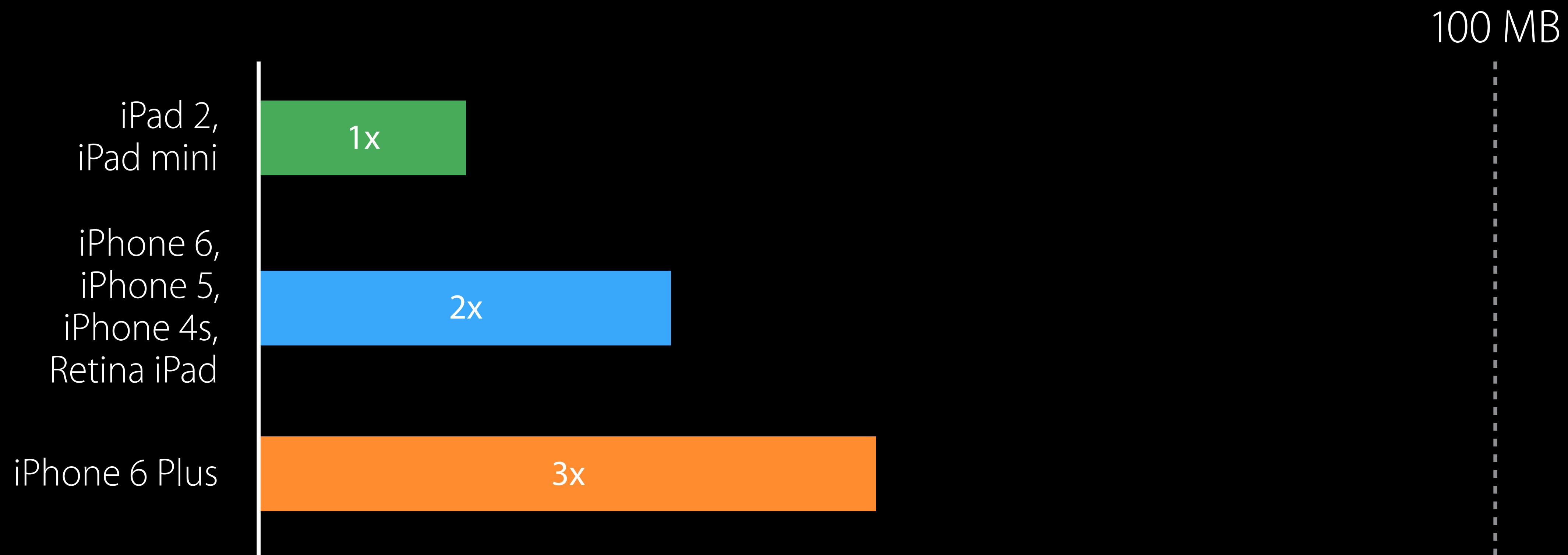
App Slicing

NEW

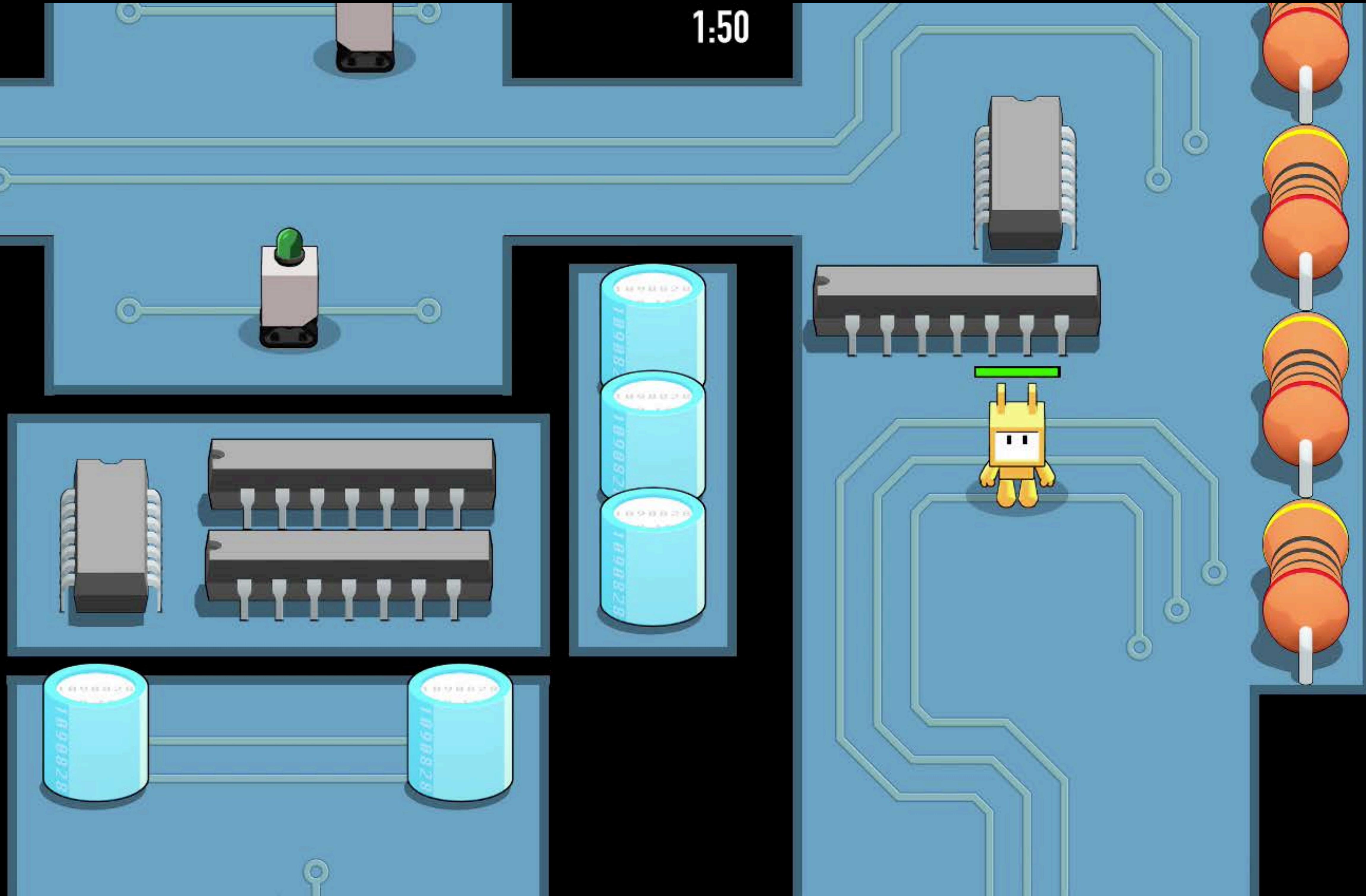


App Slicing

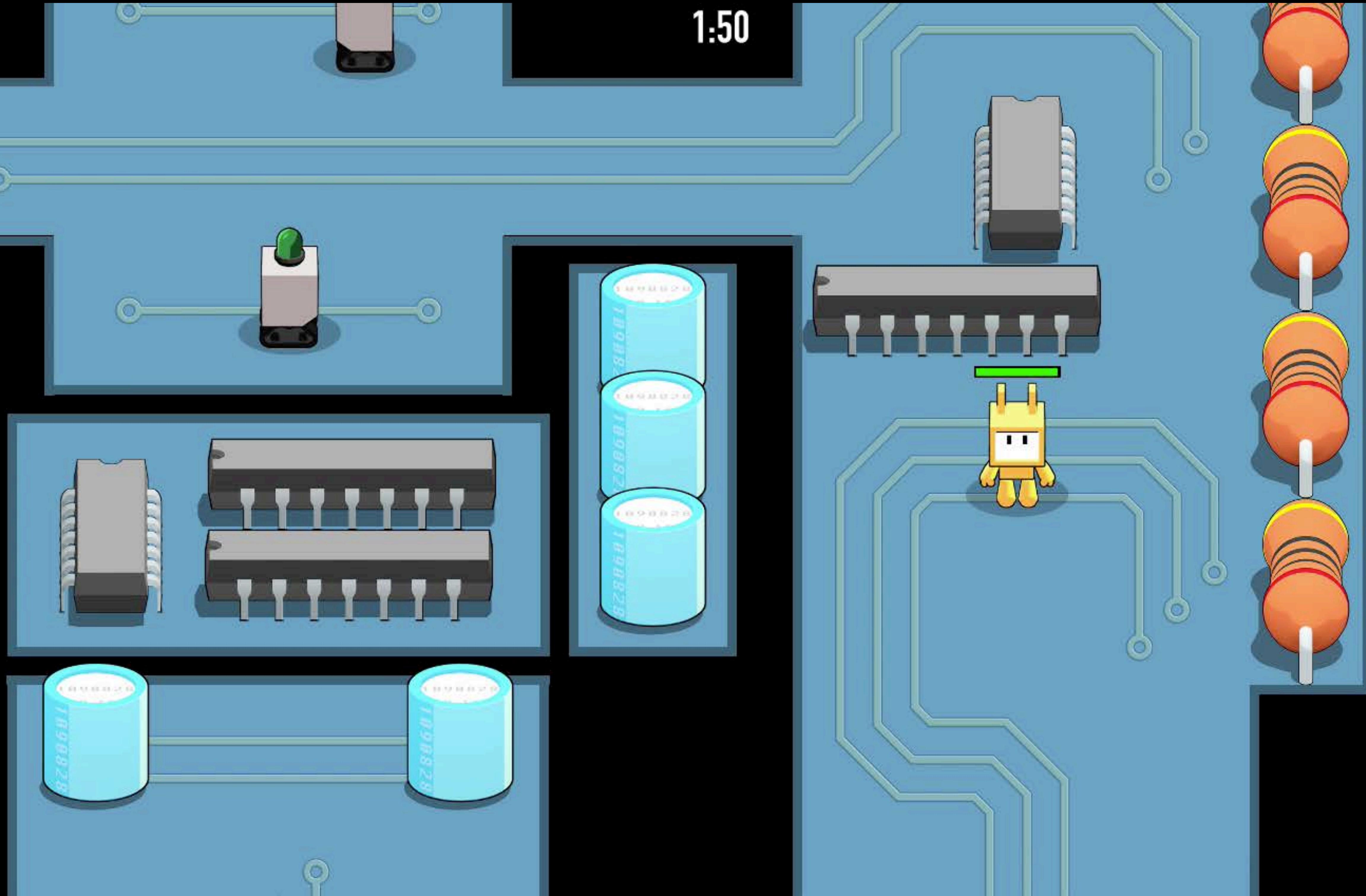
NEW



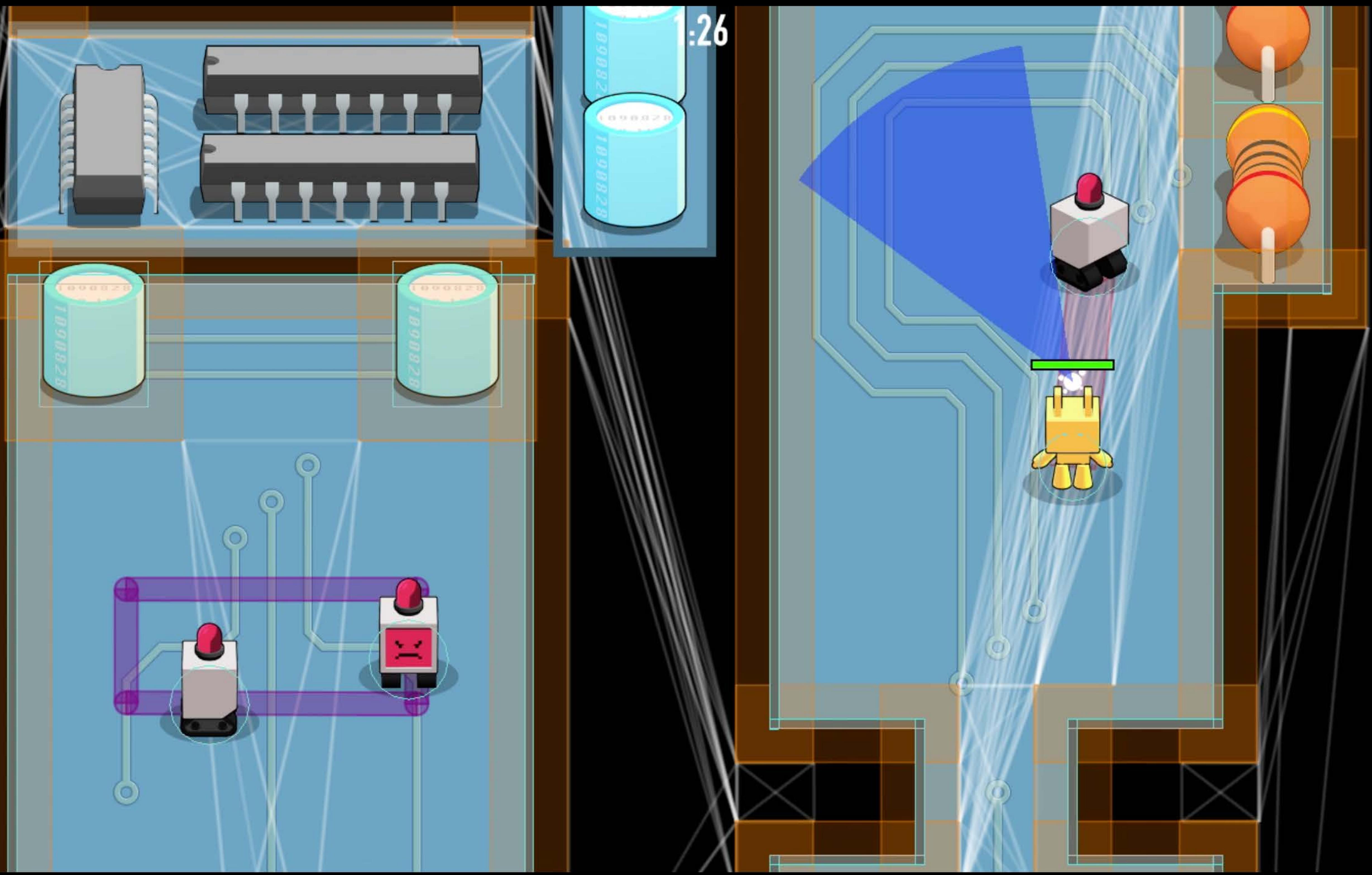
1:50



1:50

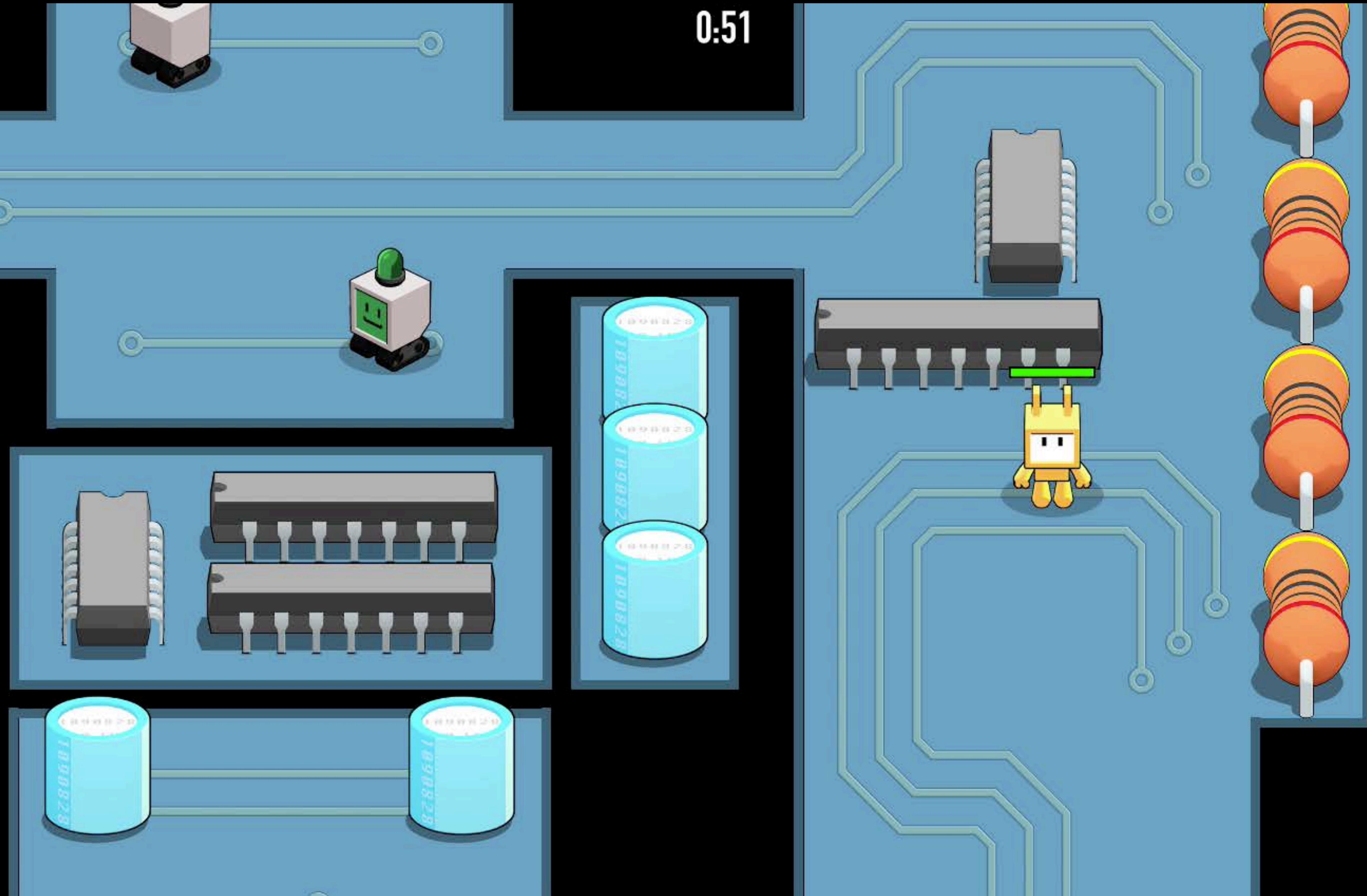


:26

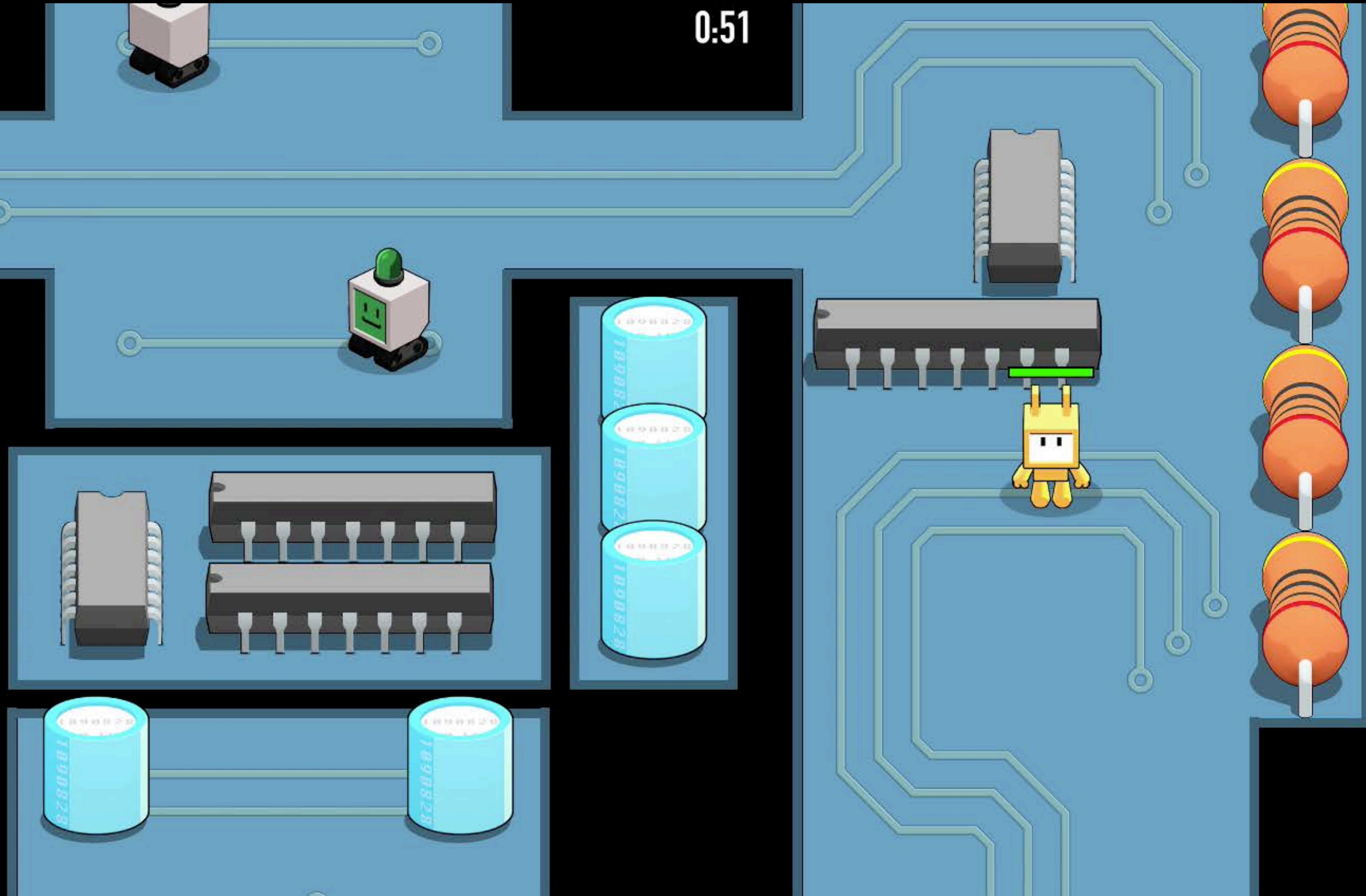


16 Orientations

0:51

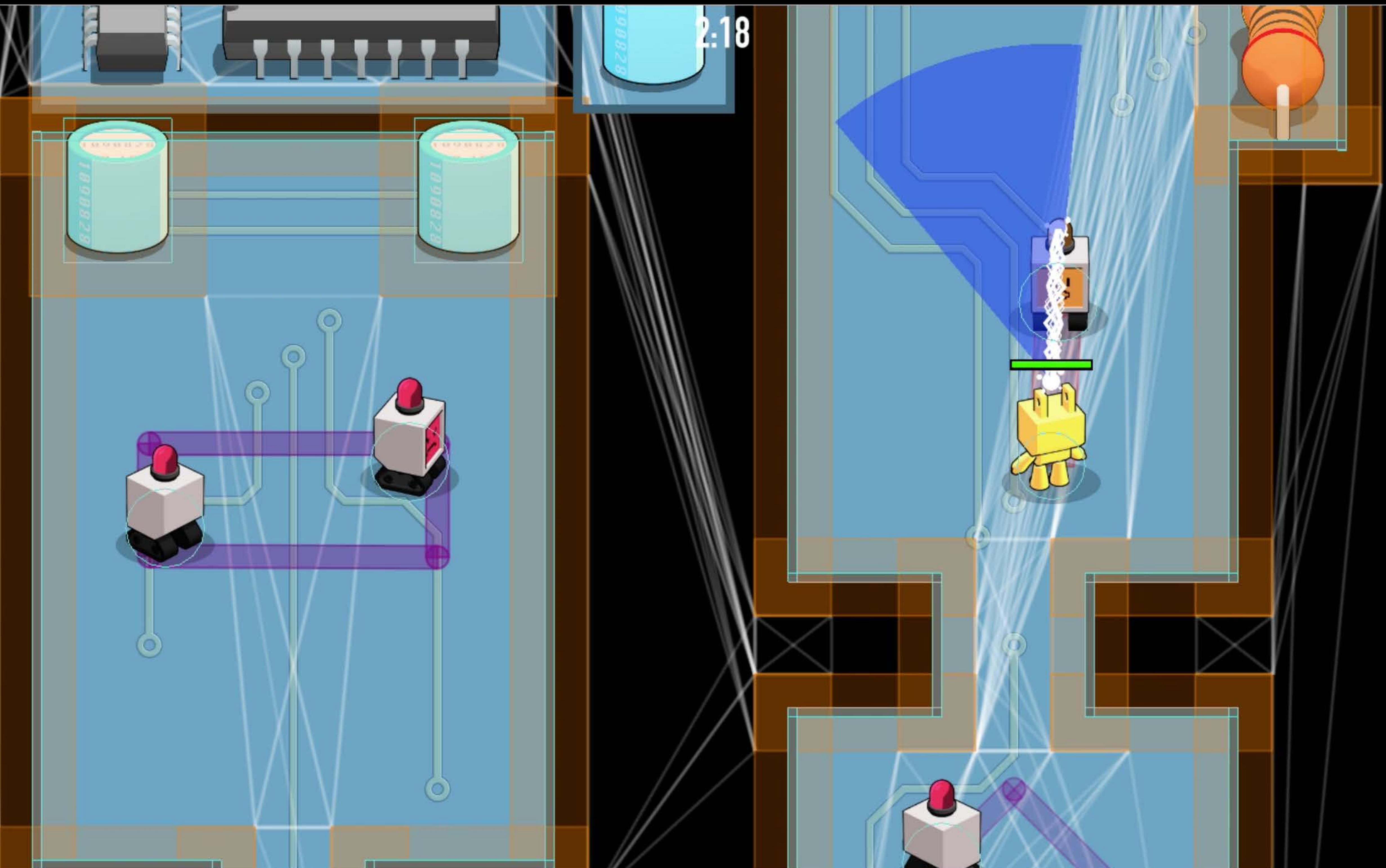


0:51



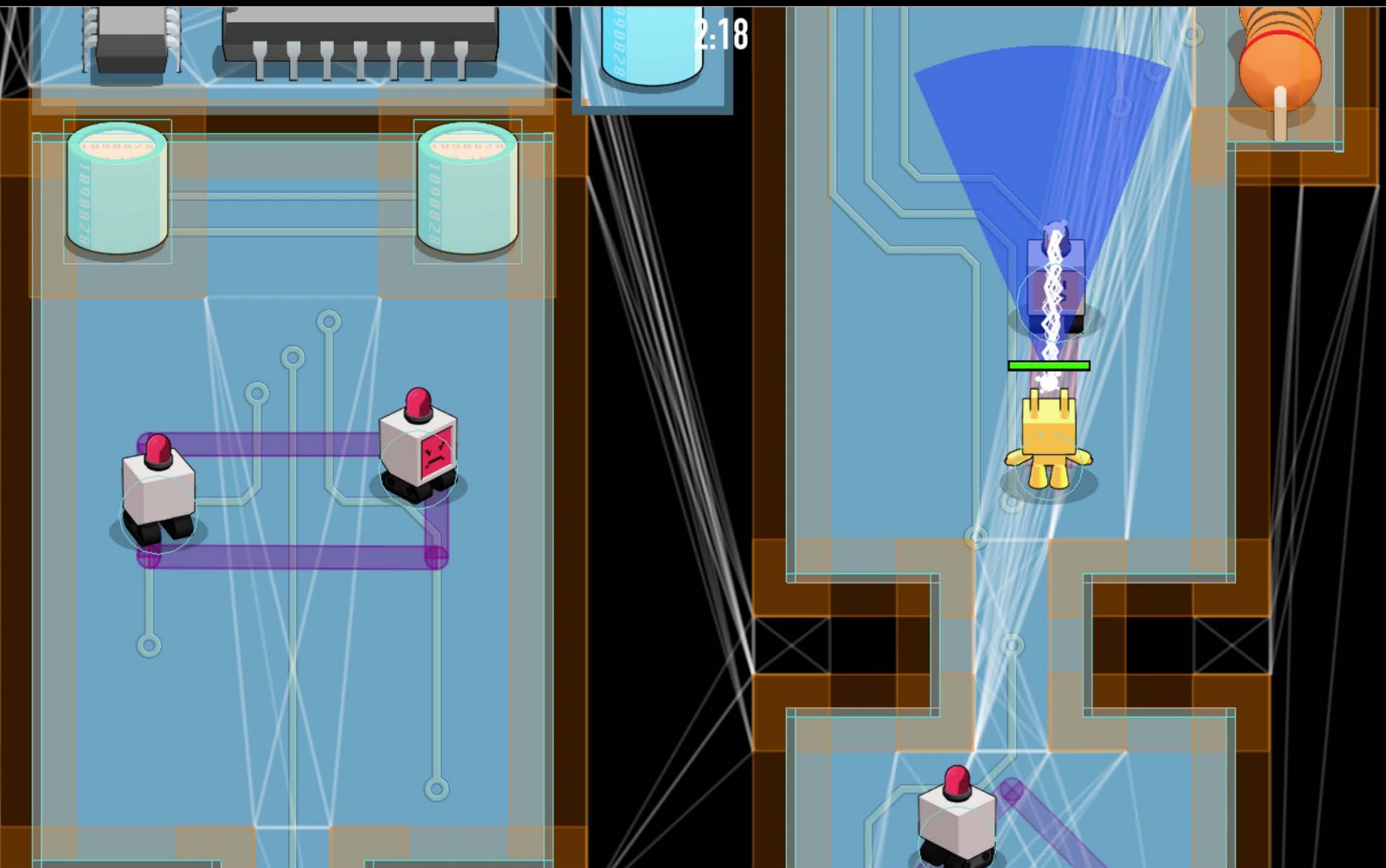
2:18

fps: 54.9



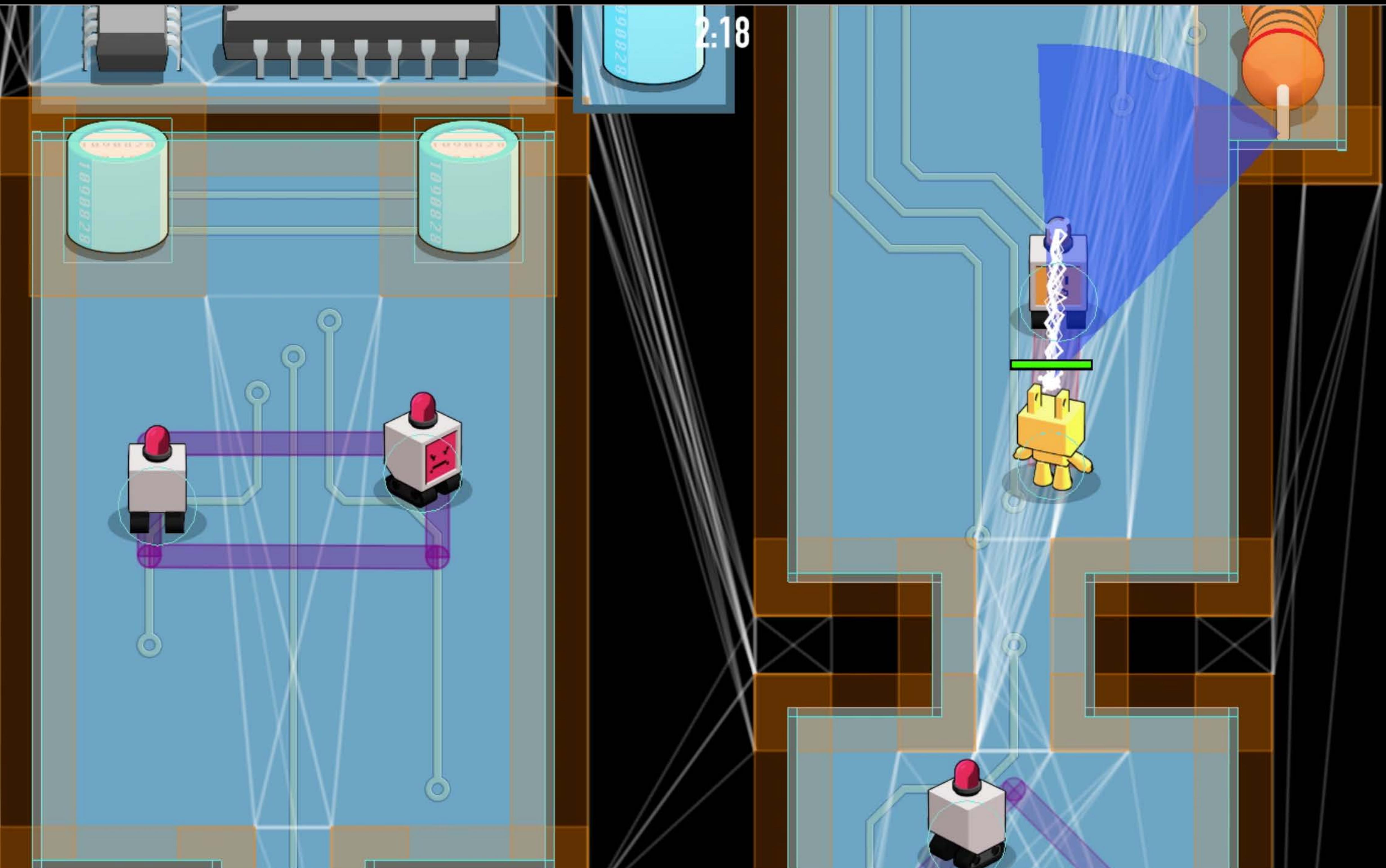
2:18

fps: 54.



2:18

fps: 54.



App Slicing

In DemoBots

App Slicing

In DemoBots

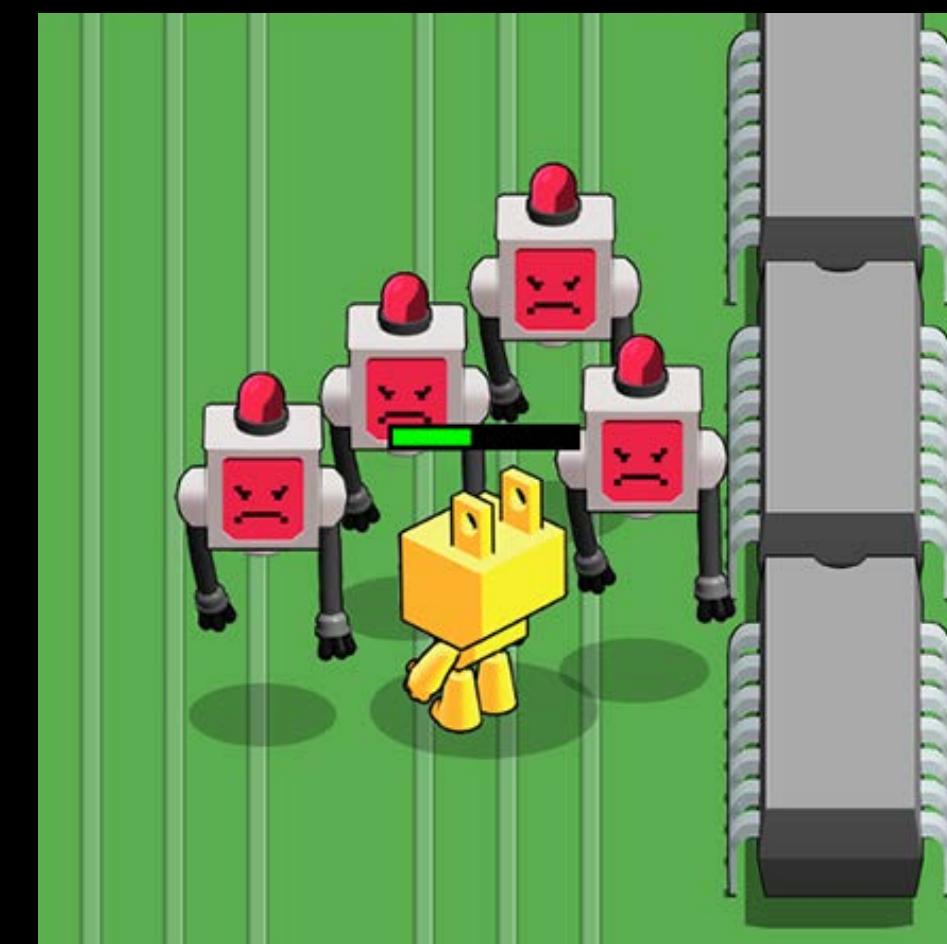
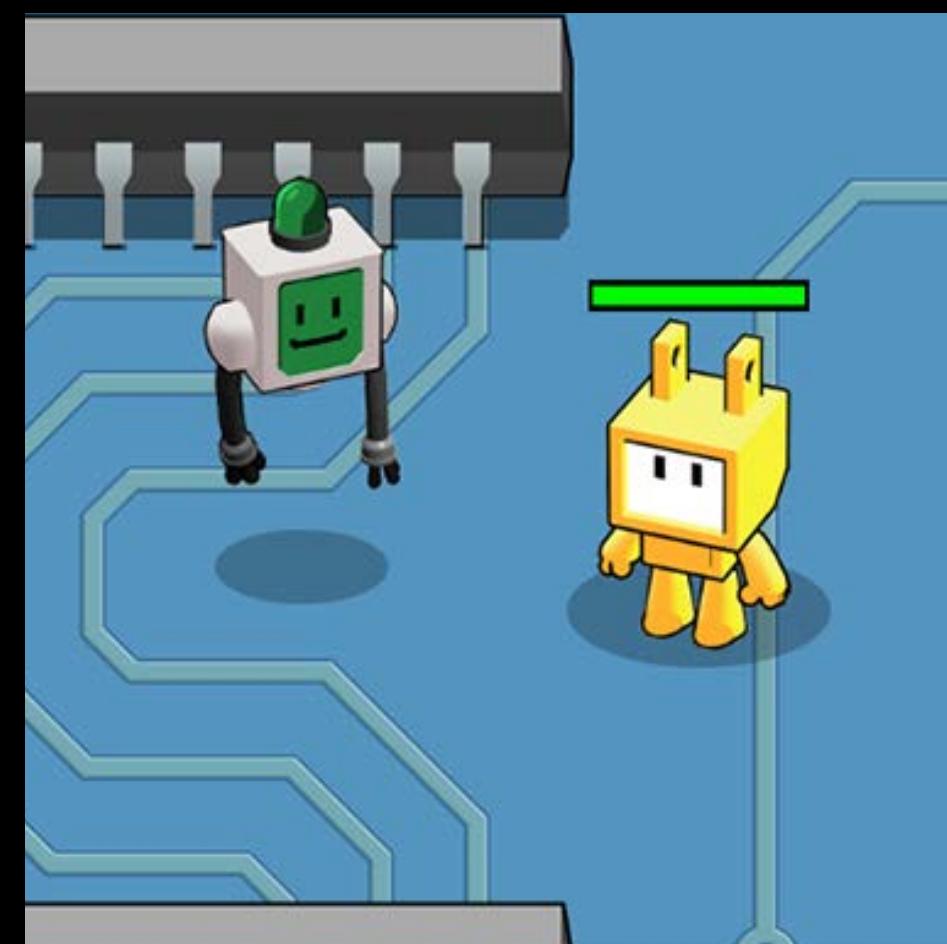
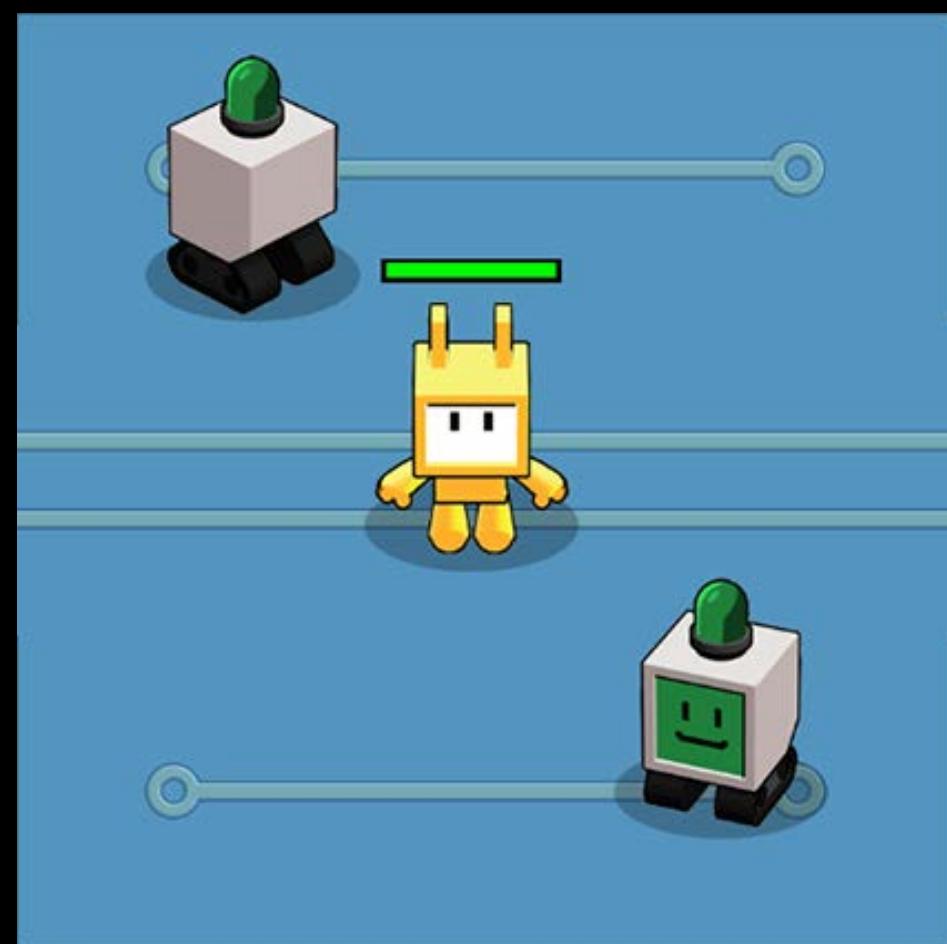
Texture atlases in asset catalogs

- Decrease the app size
- Improve gameplay with 16-way animations

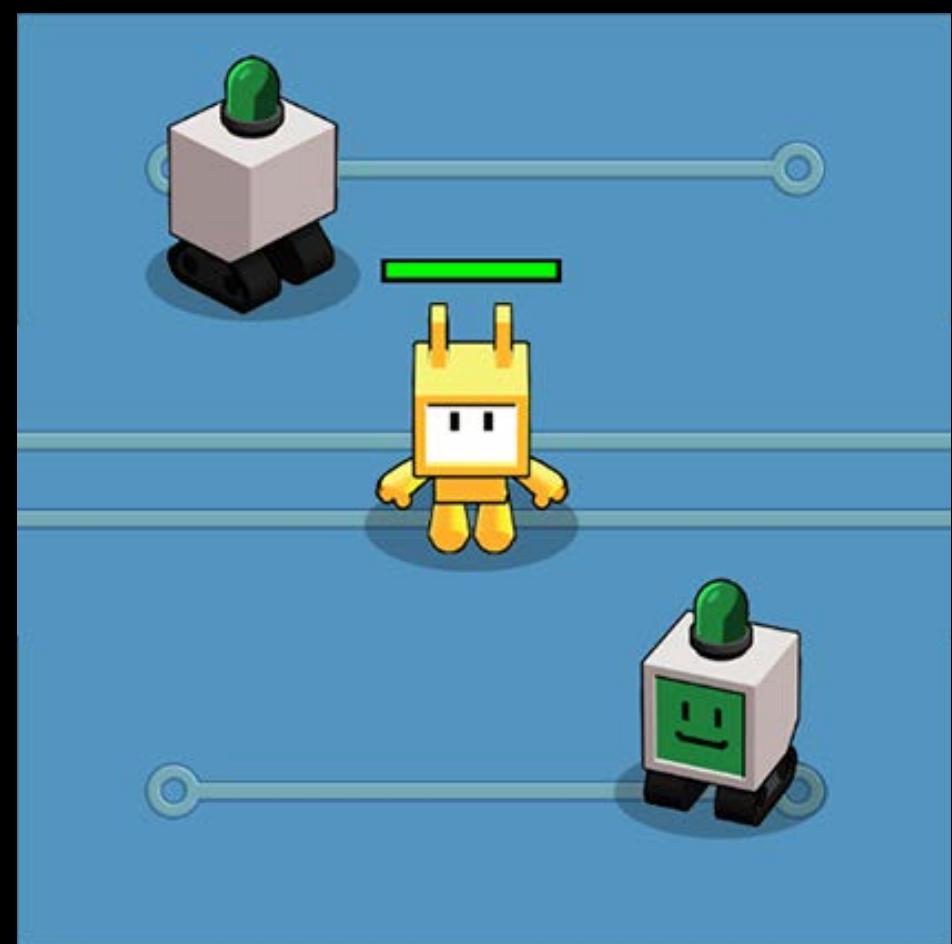
NEW

On Demand Resources

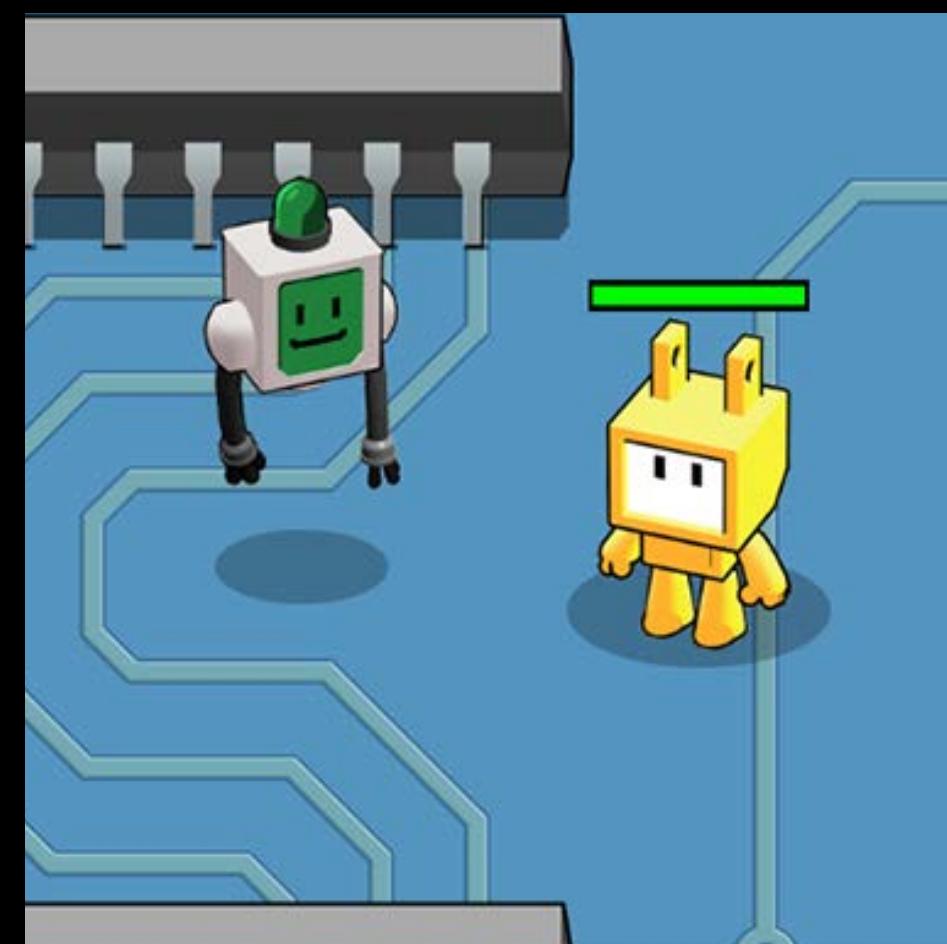
Level Tags



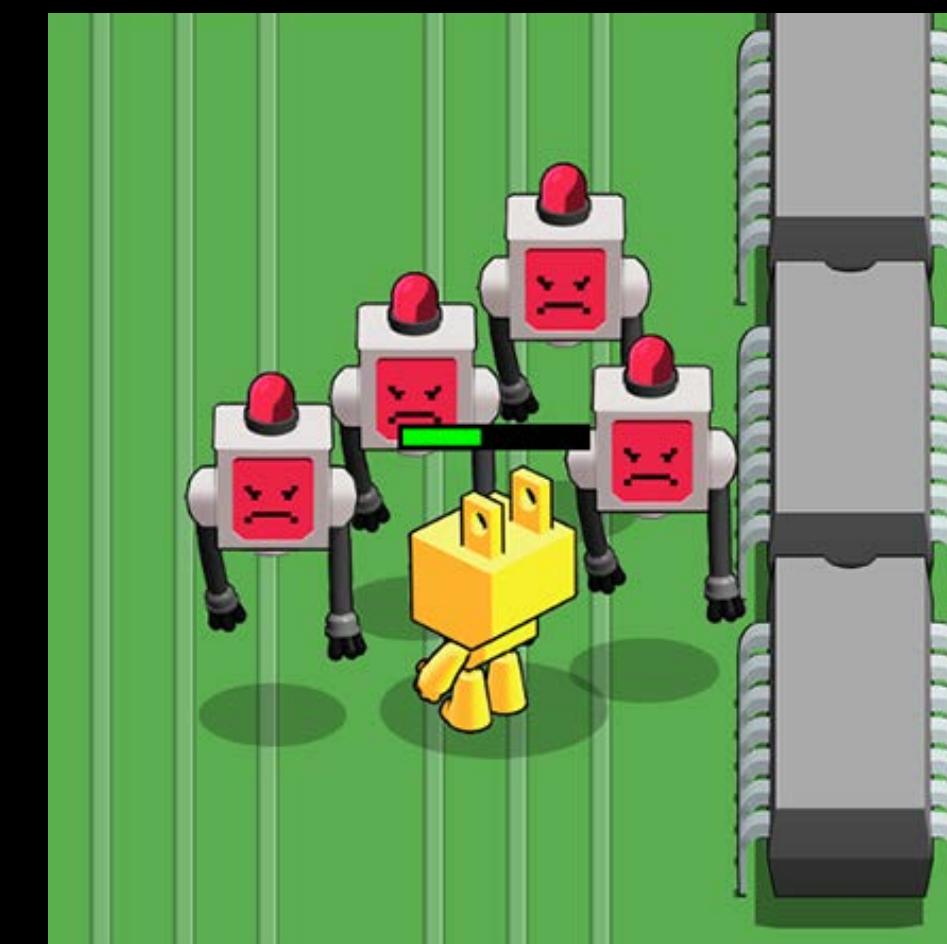
Level Tags



Level1

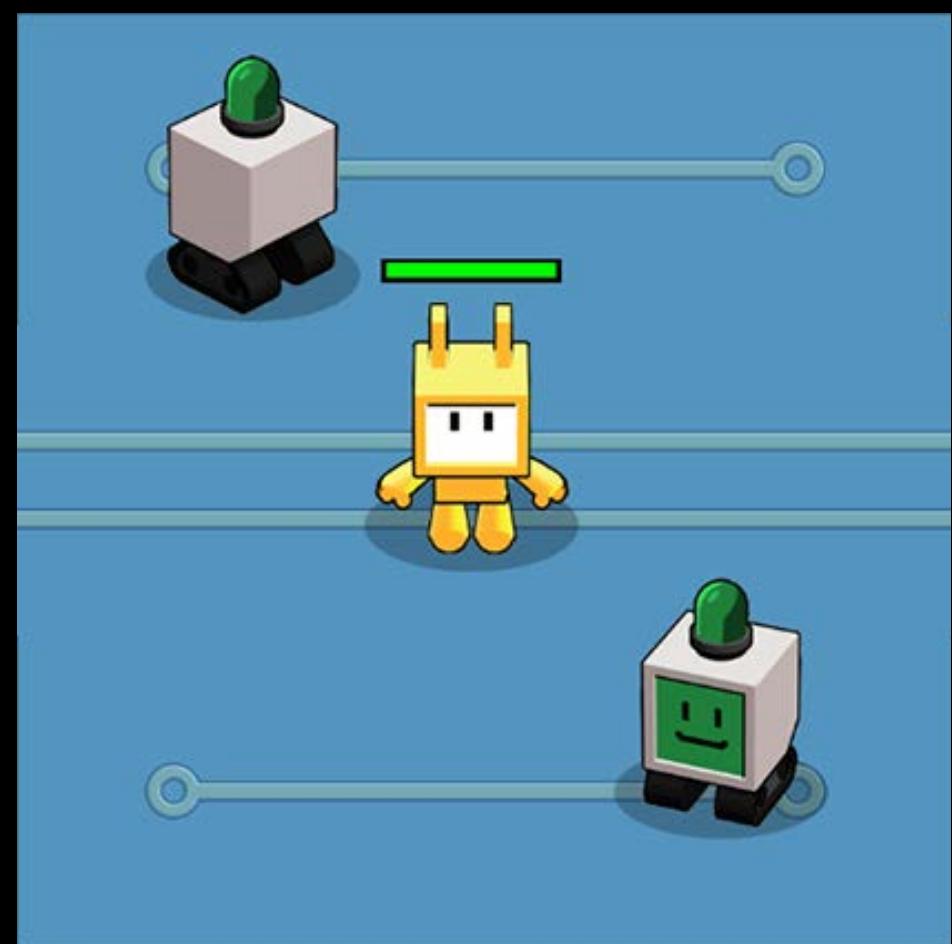


Level2

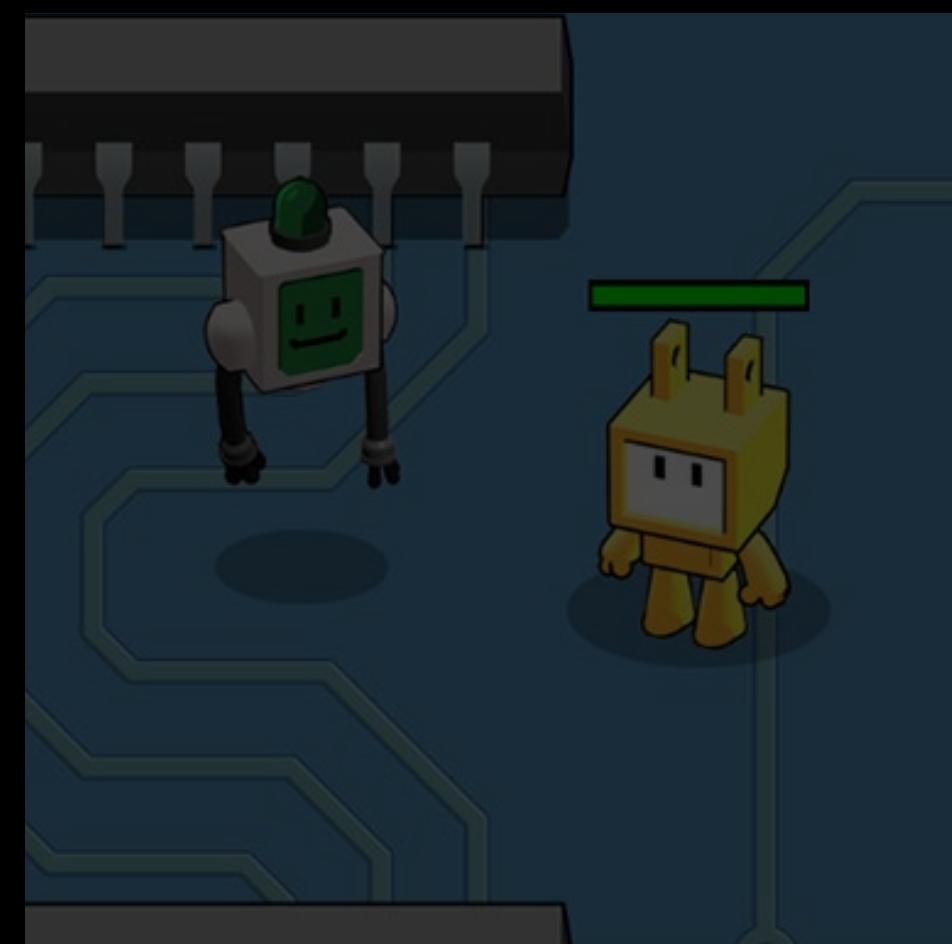


Level3

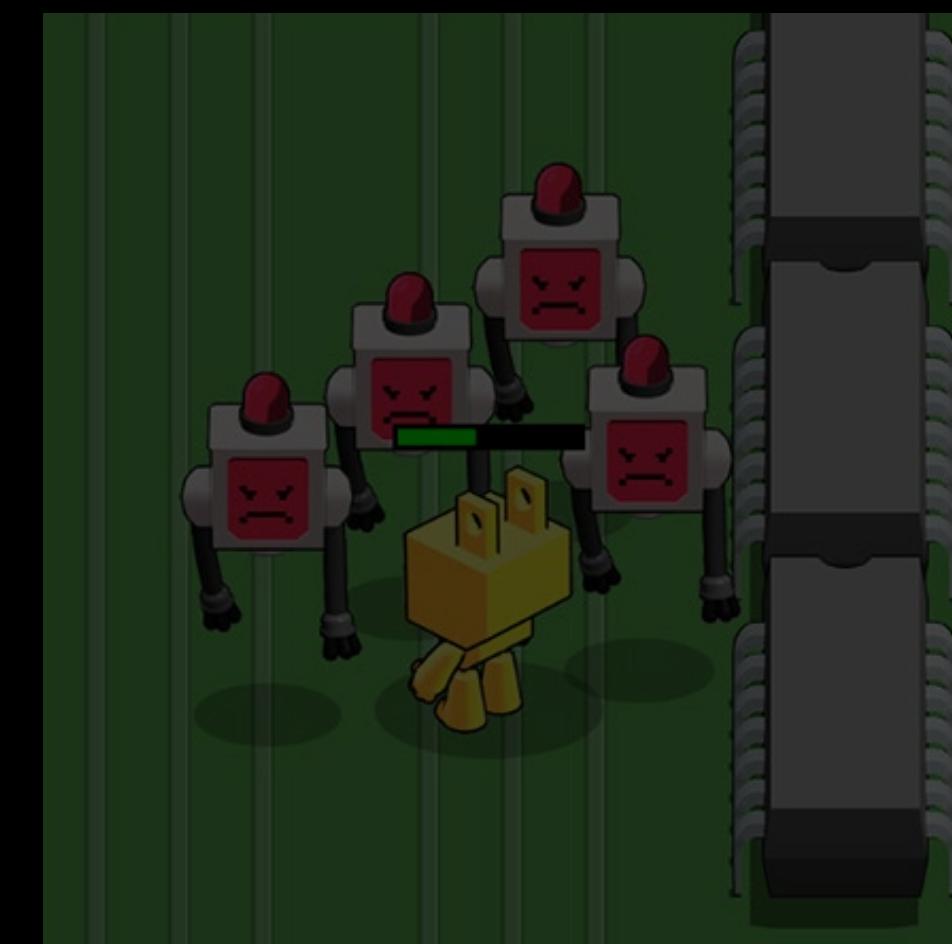
Level Tags



Level1

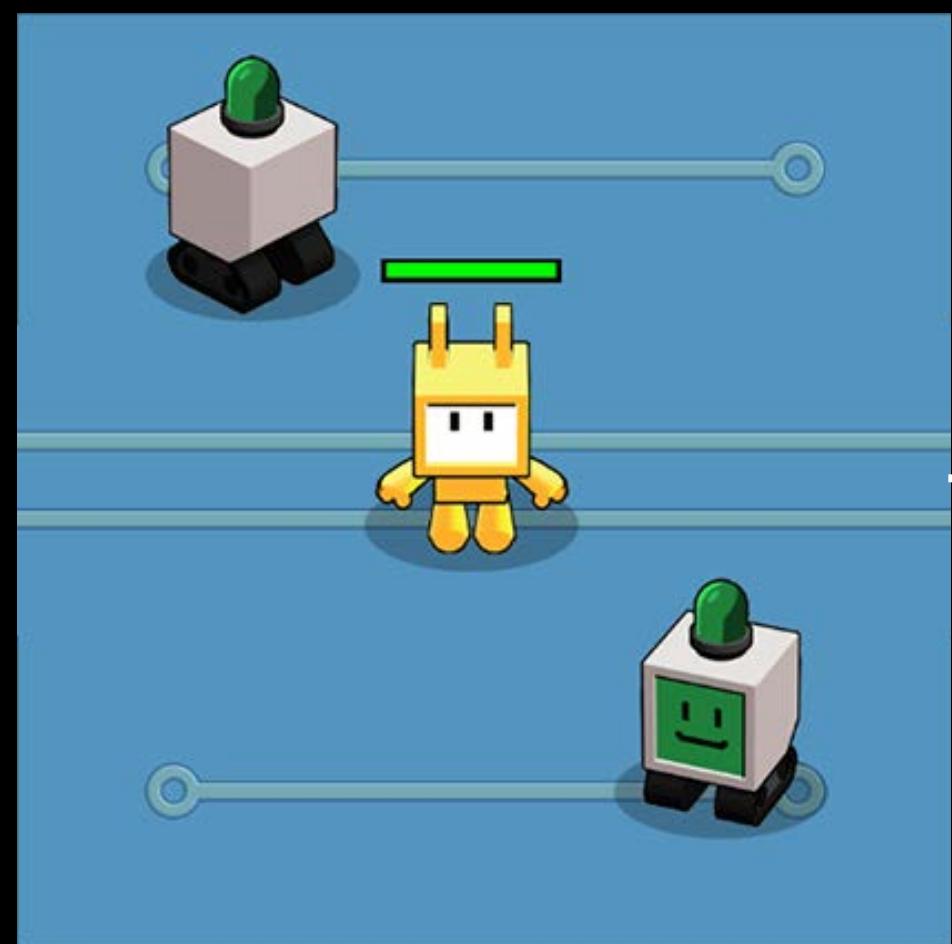


Level2

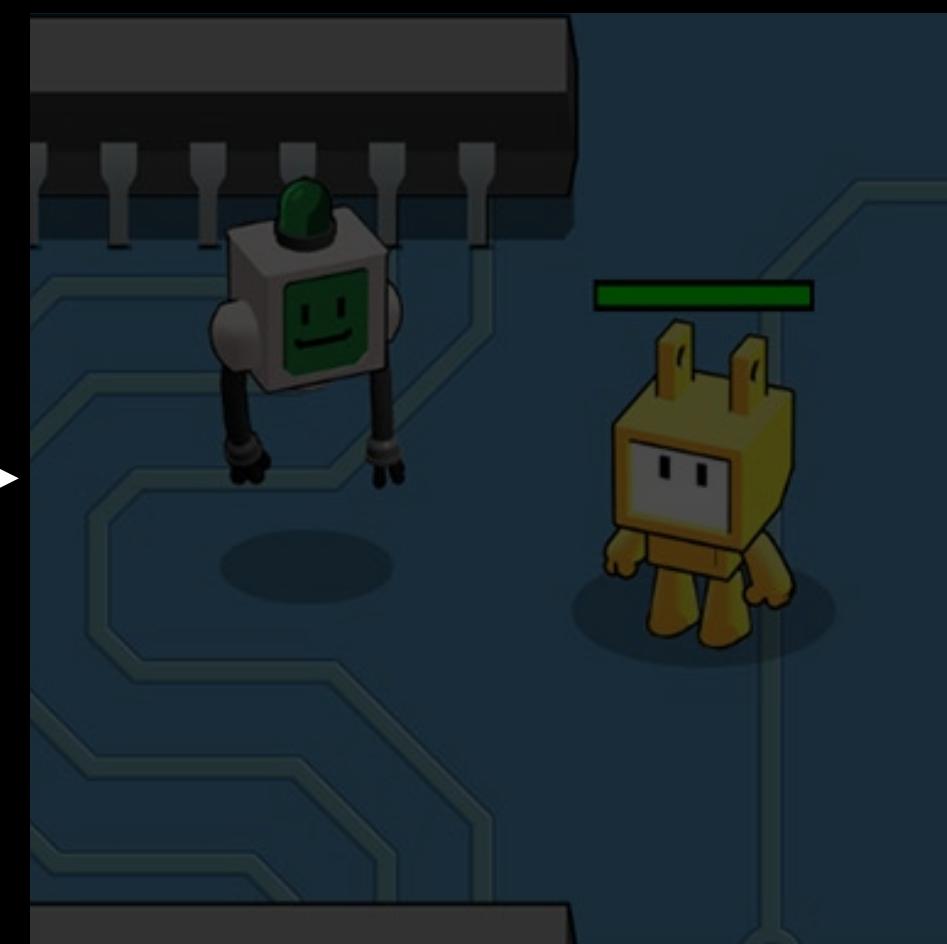


Level3

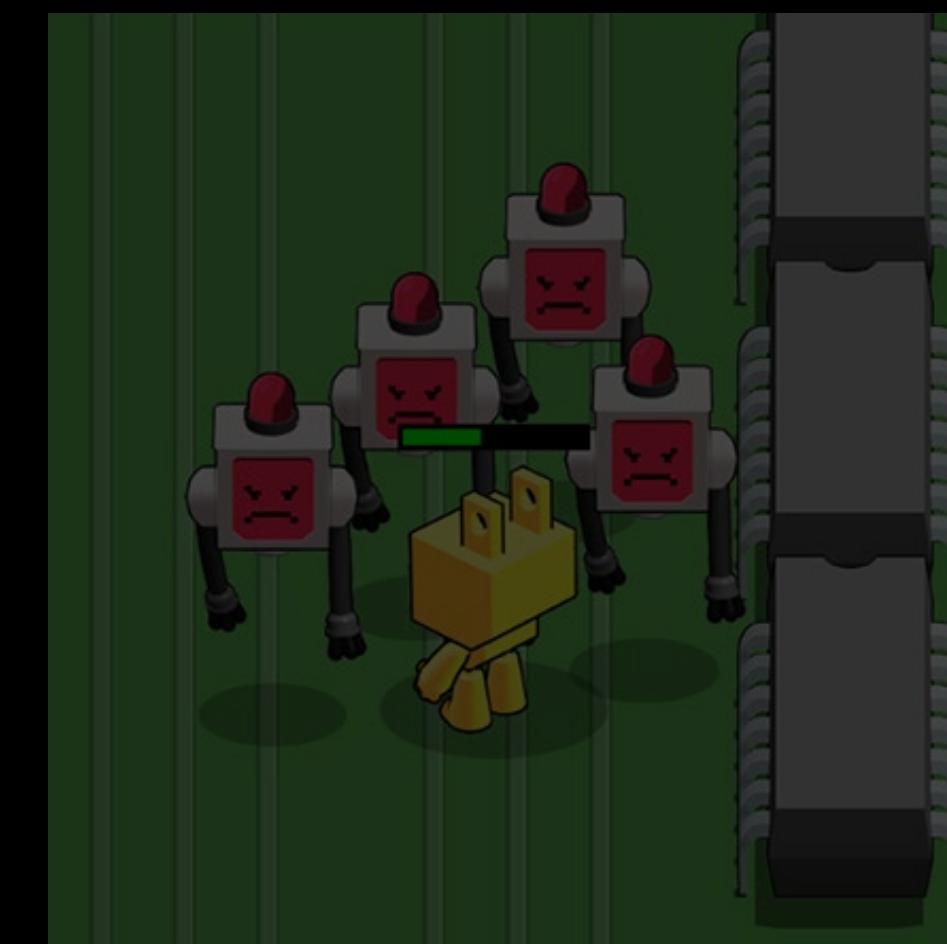
Level Tags



Level1

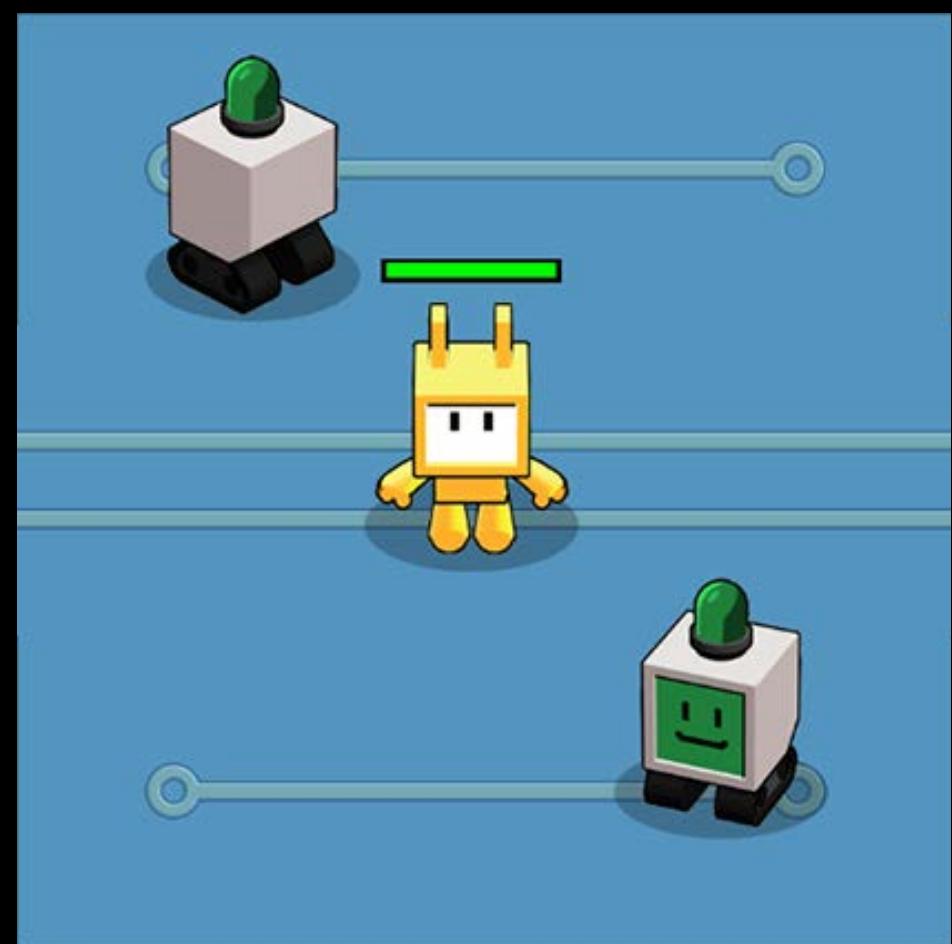


Level2

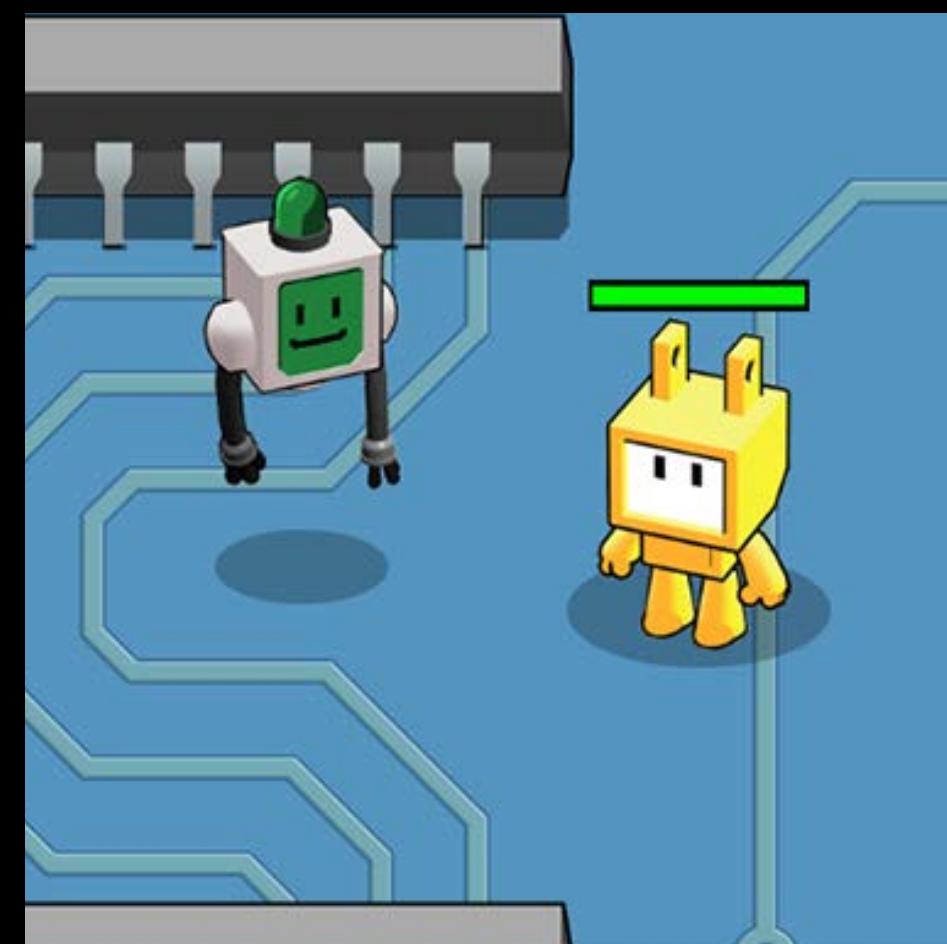


Level3

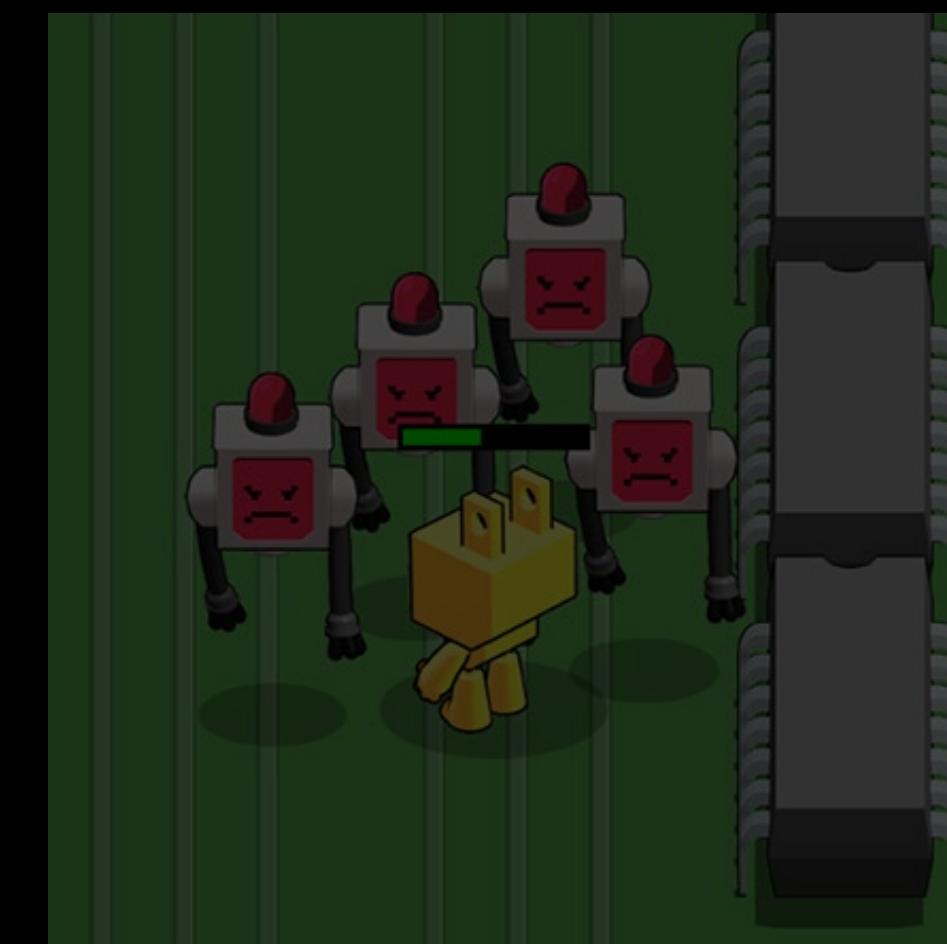
Level Tags



Level1

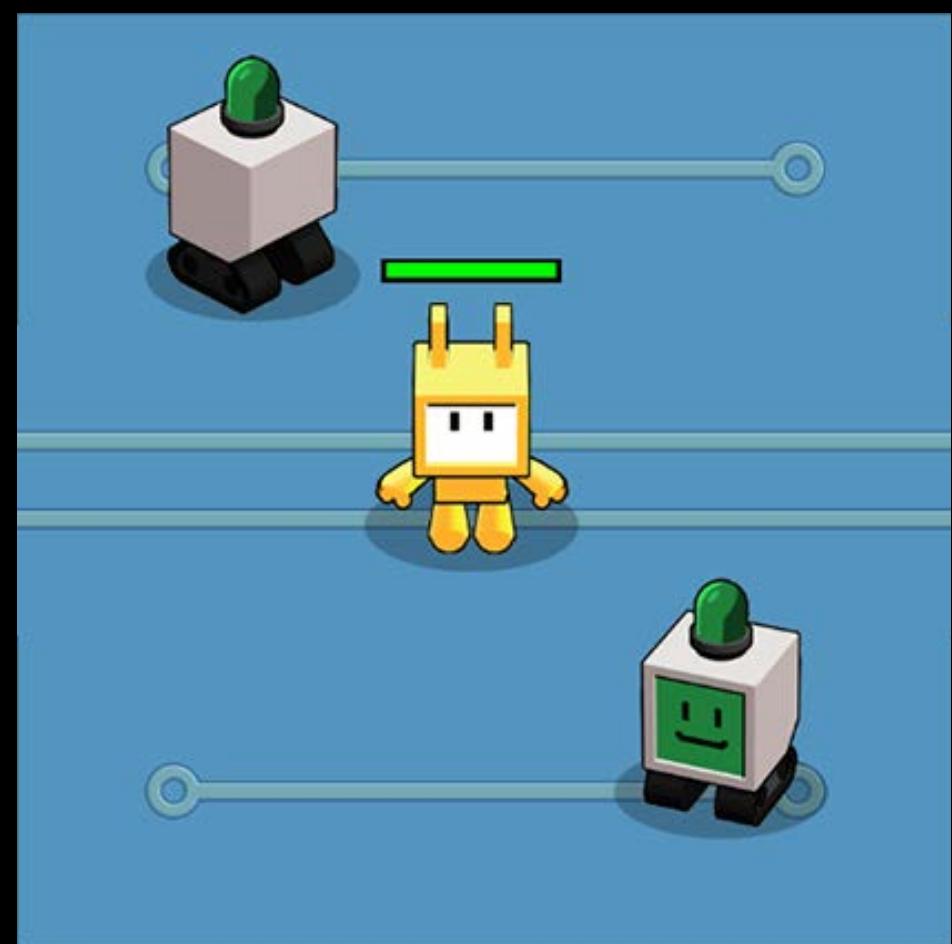


Level2

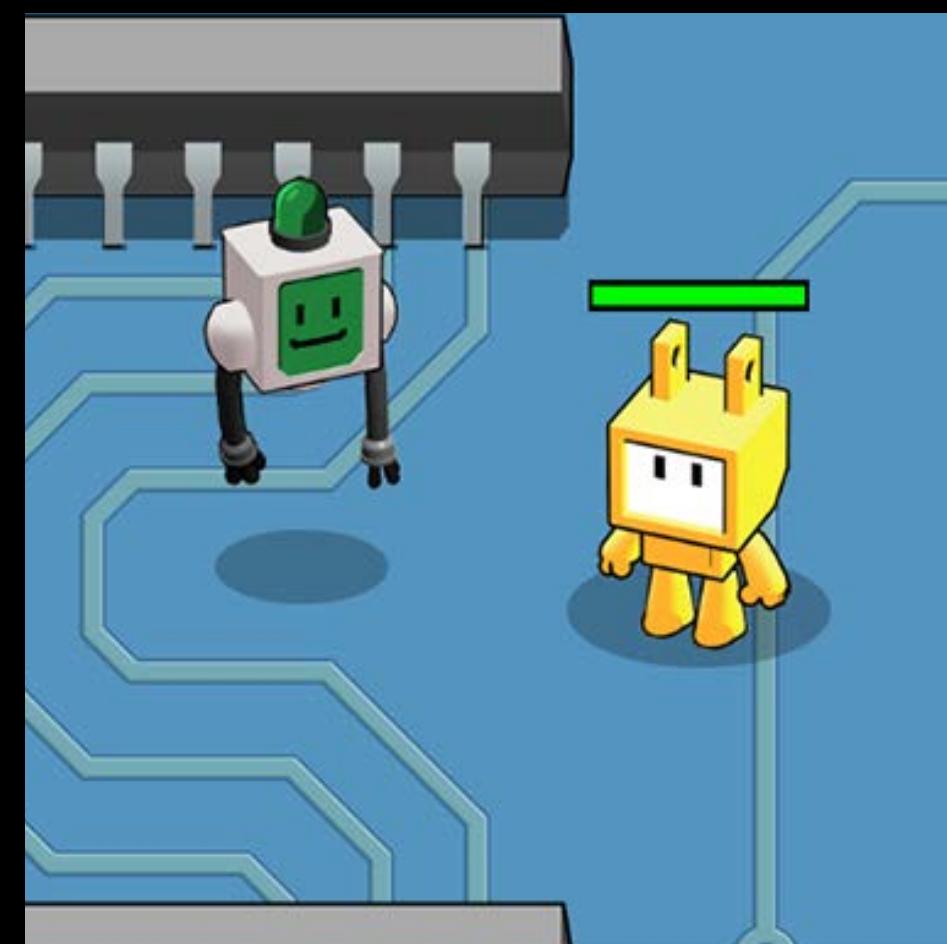


Level3

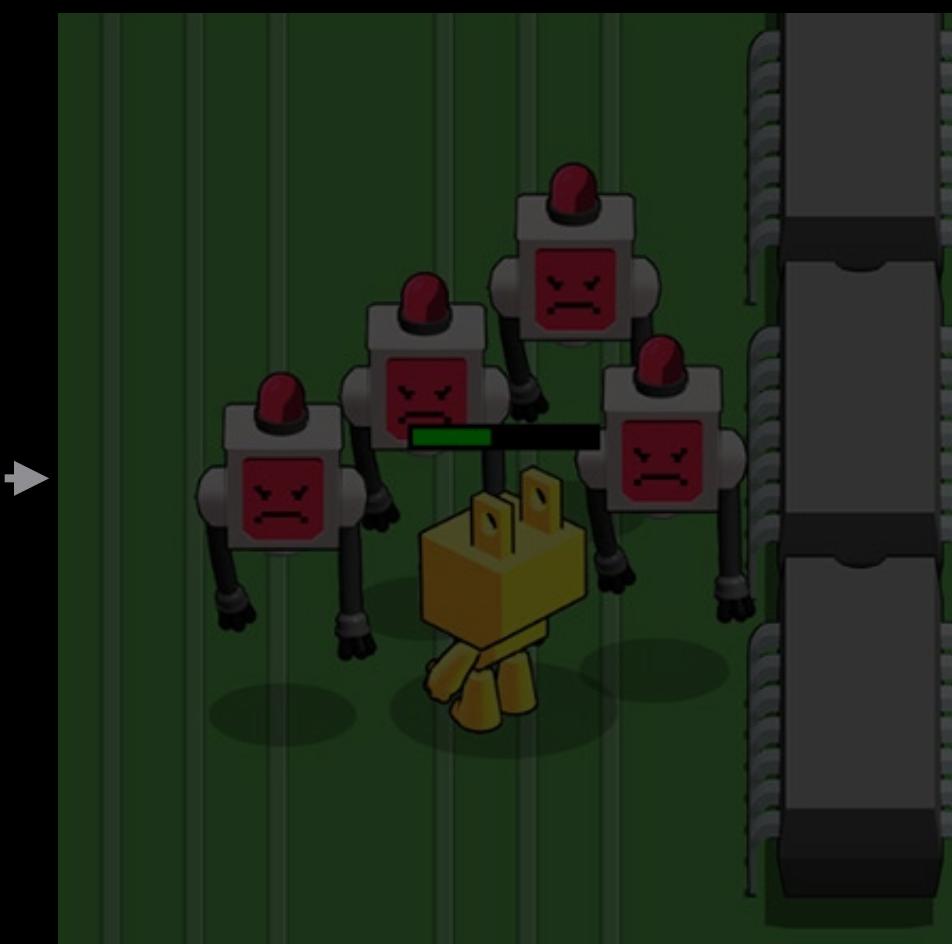
Level Tags



Level1

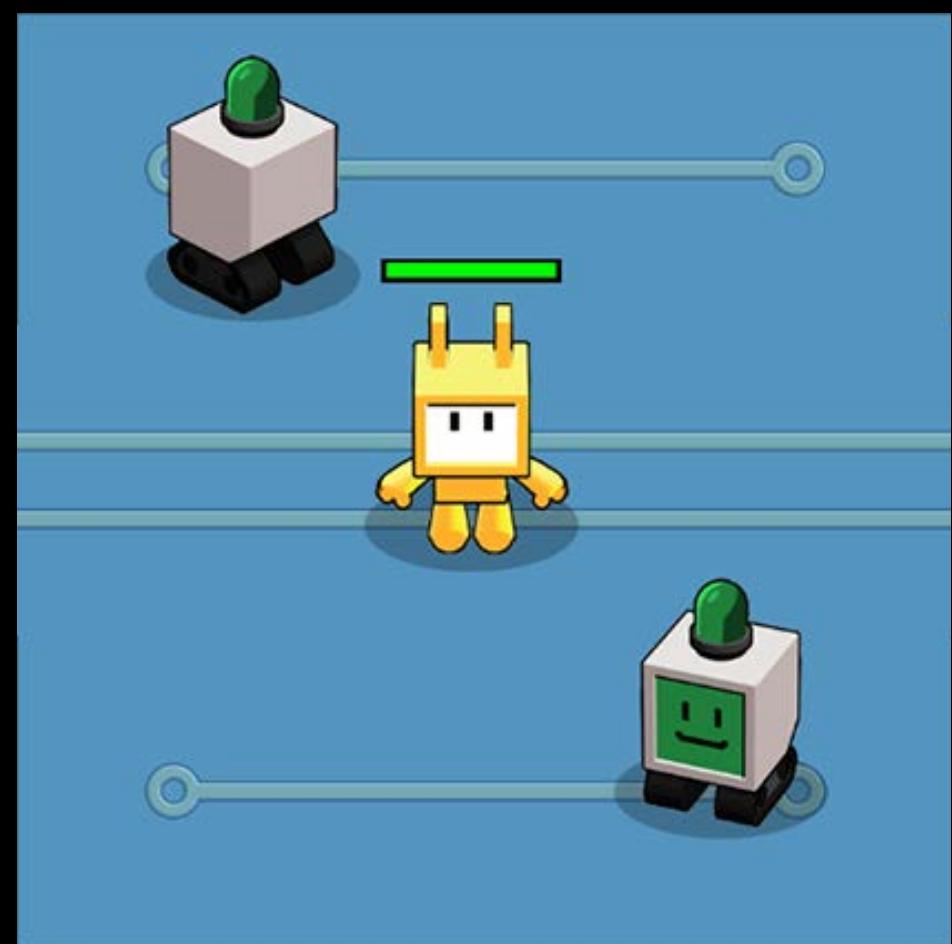


Level2

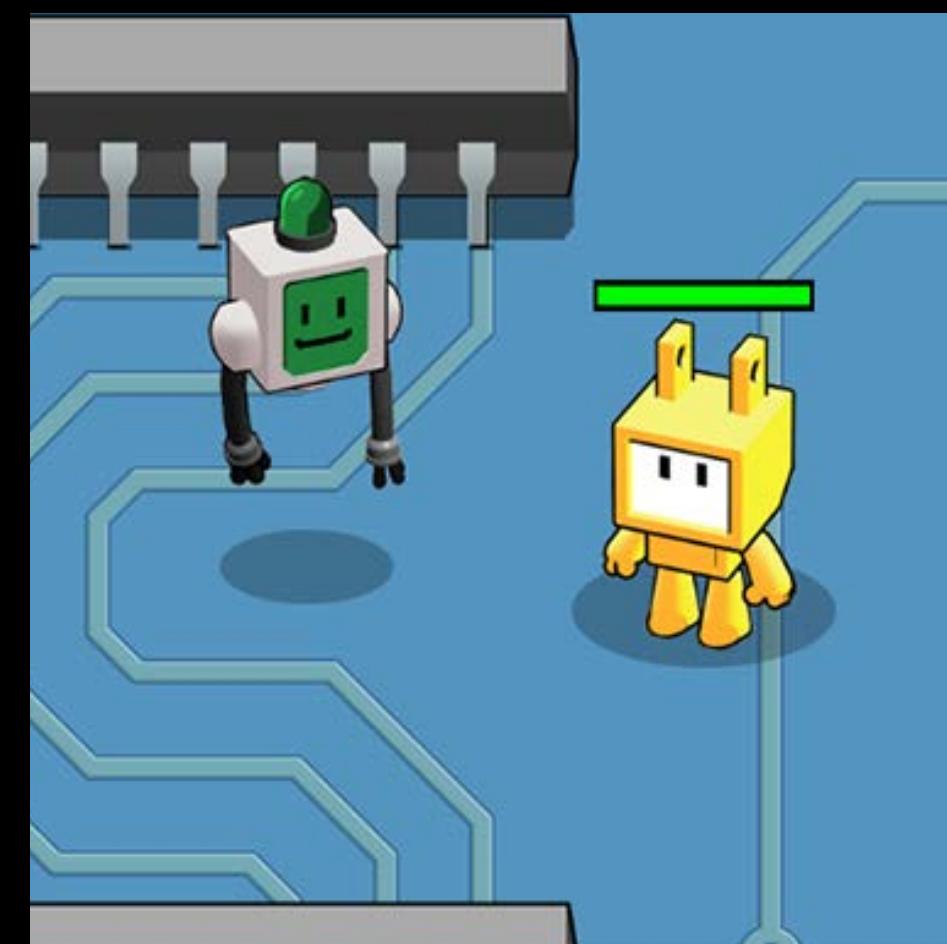


Level3

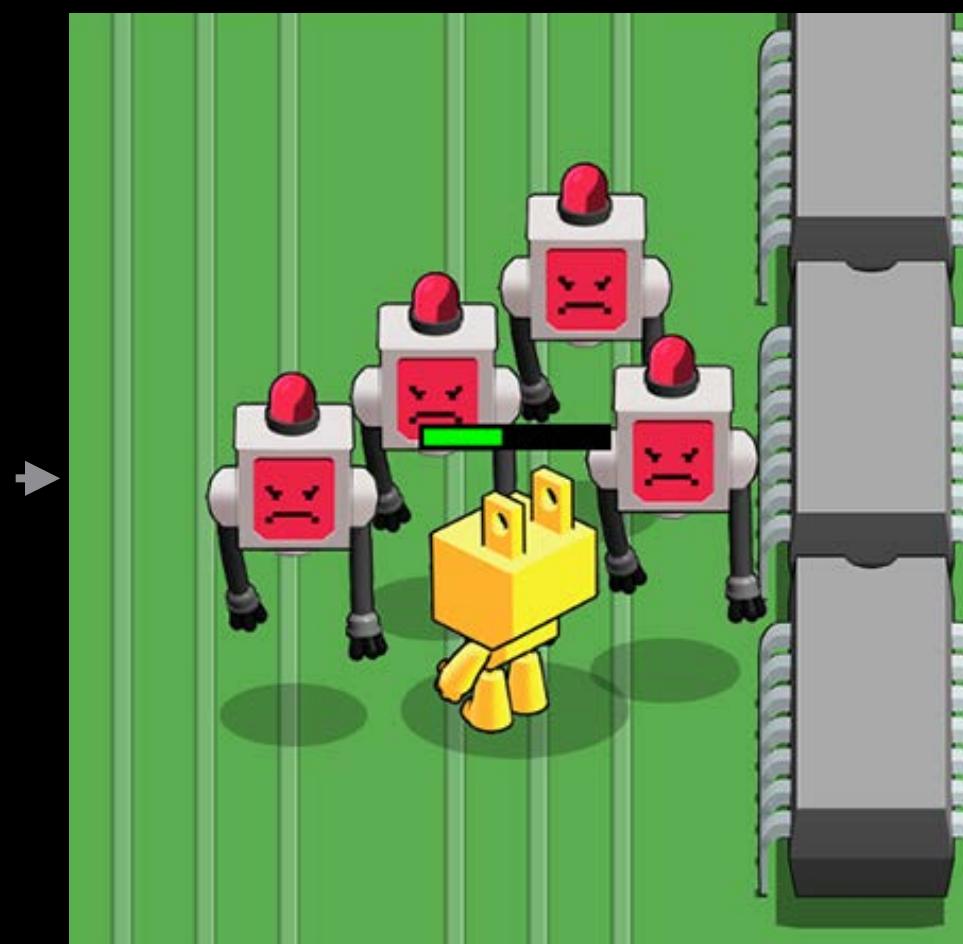
Level Tags



Level1

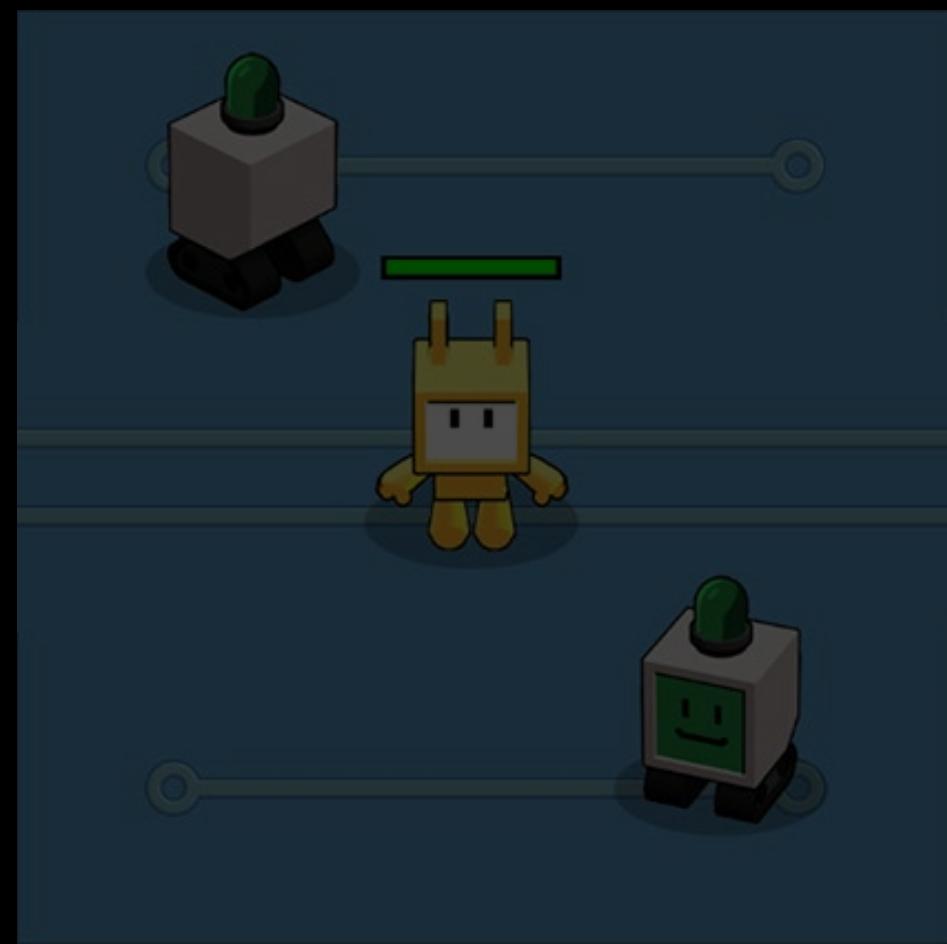


Level2

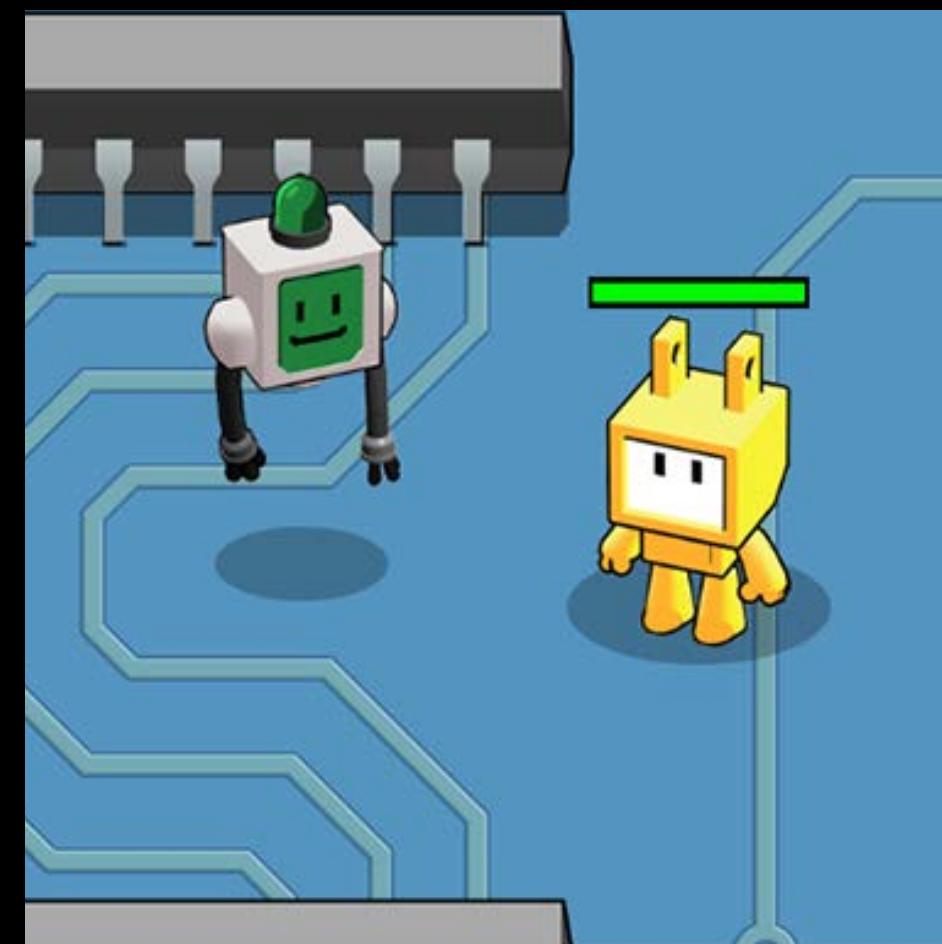


Level3

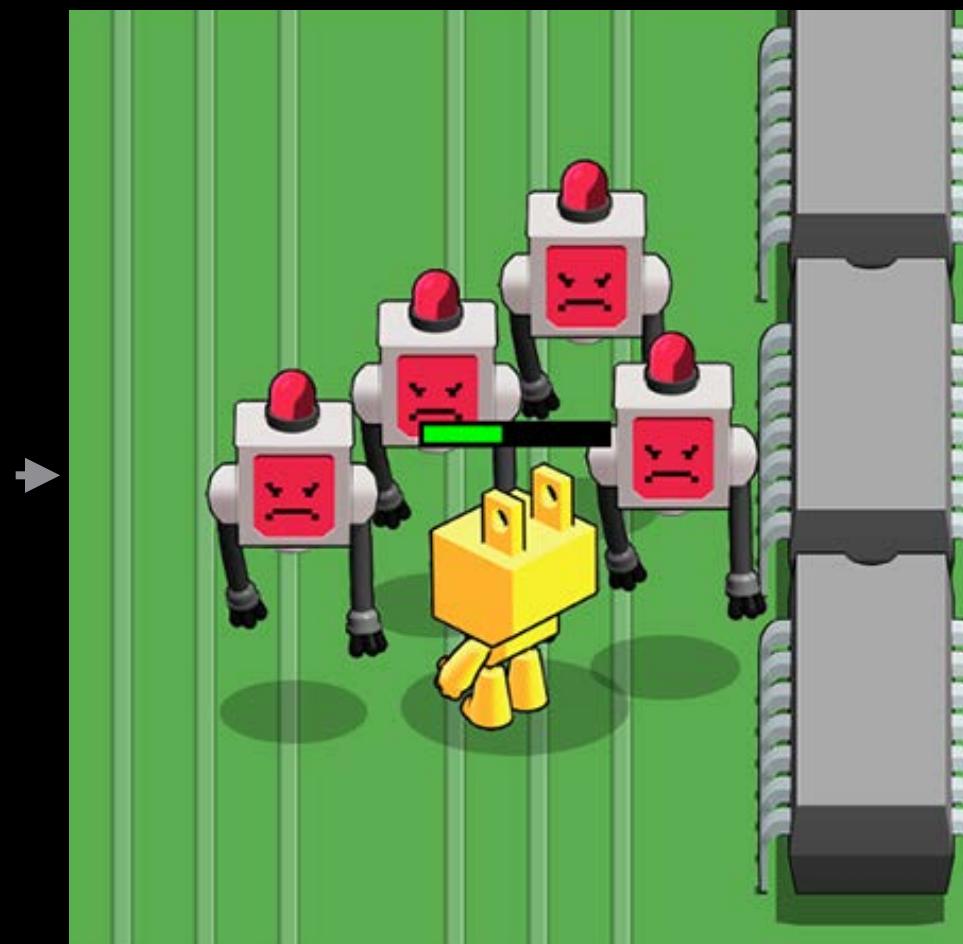
Level Tags



Level1



Level2



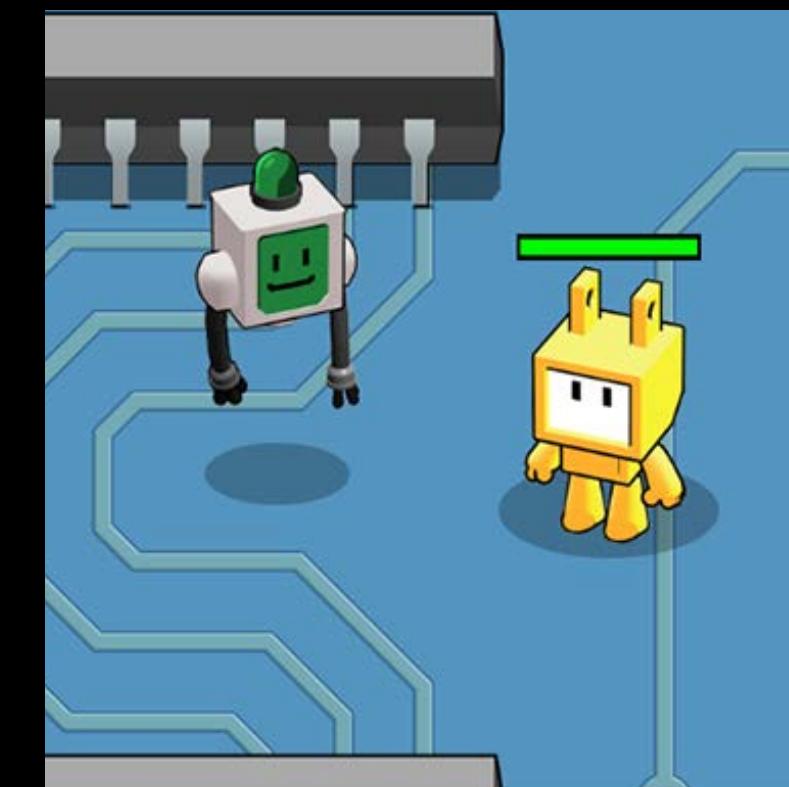
Level3

Level Tags

Level1

Level2

Level3

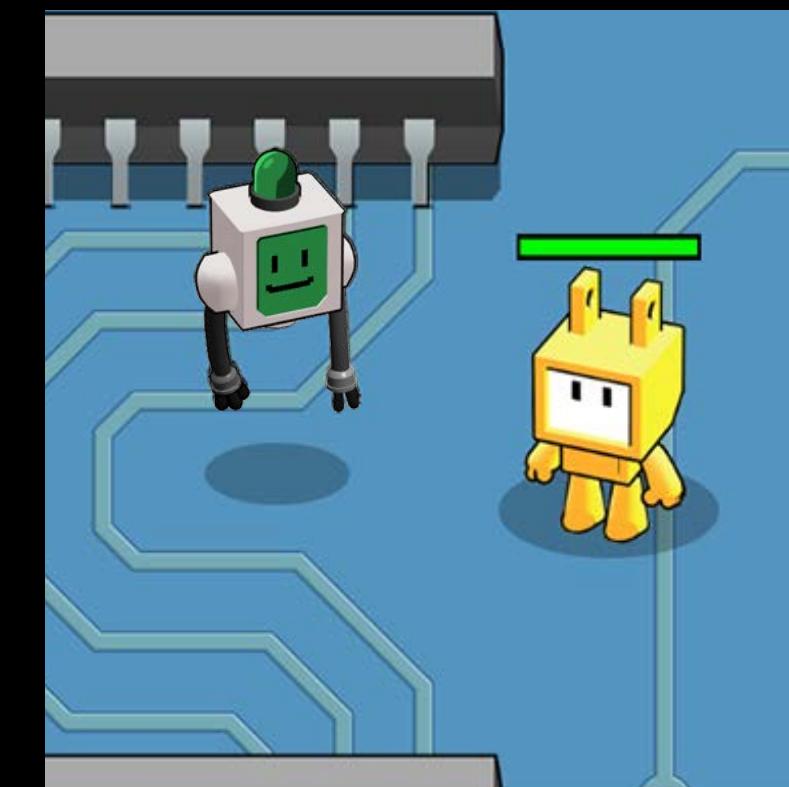


Level Tags

Level1

Level2

Level3



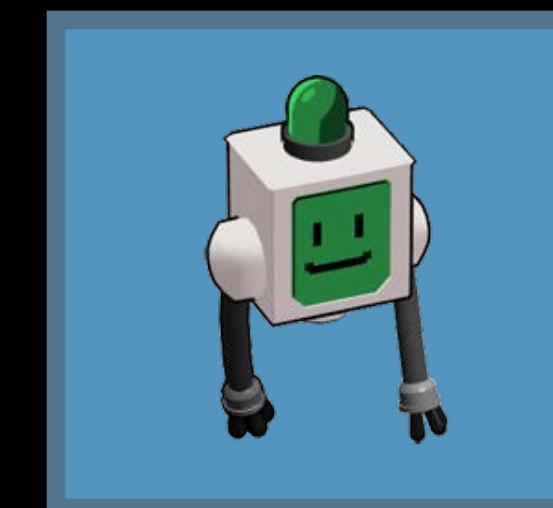
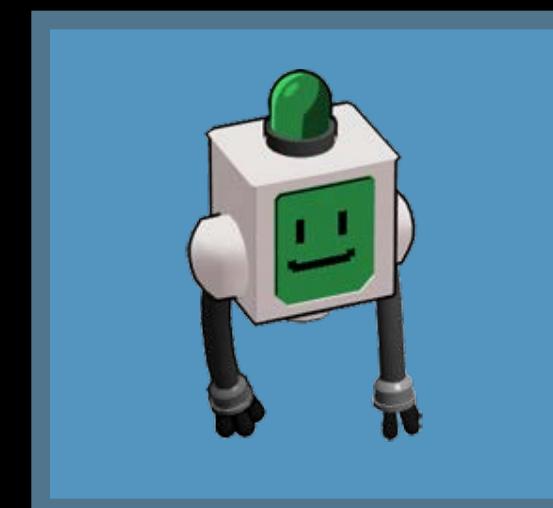
Level Tags

Level1

Level2

Level3

FlyingBot



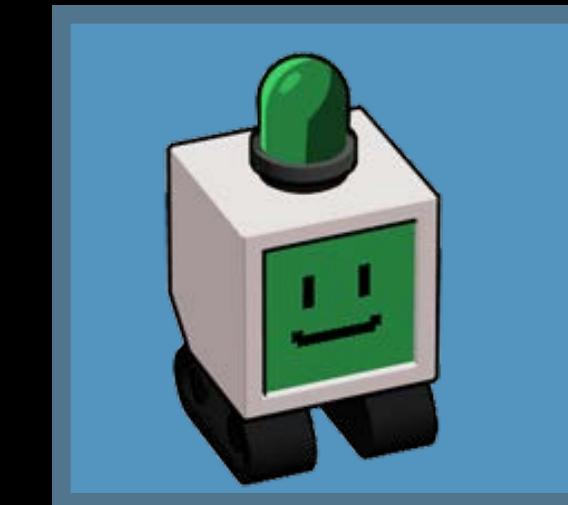
Level Tags

Level1

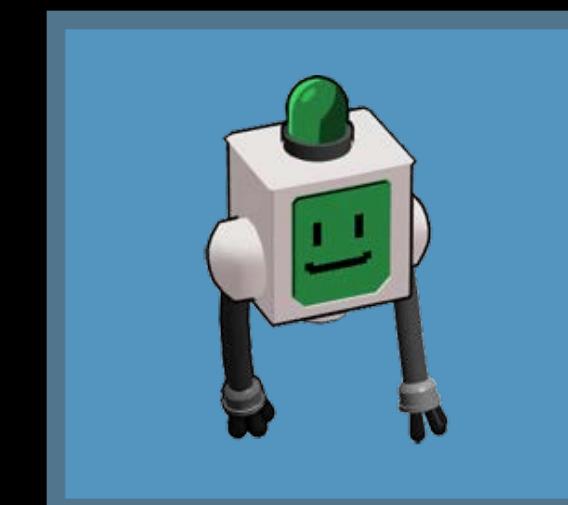
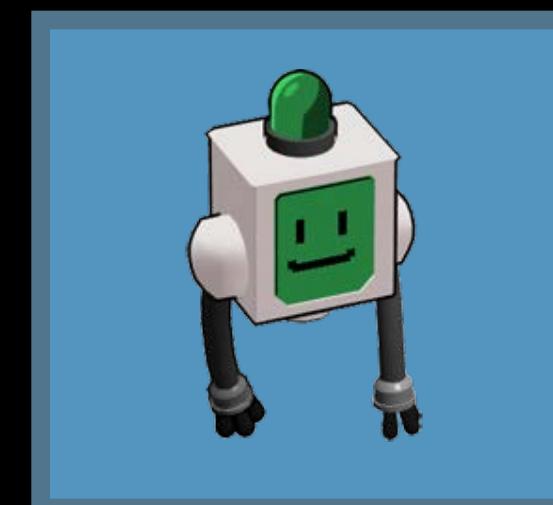
Level2

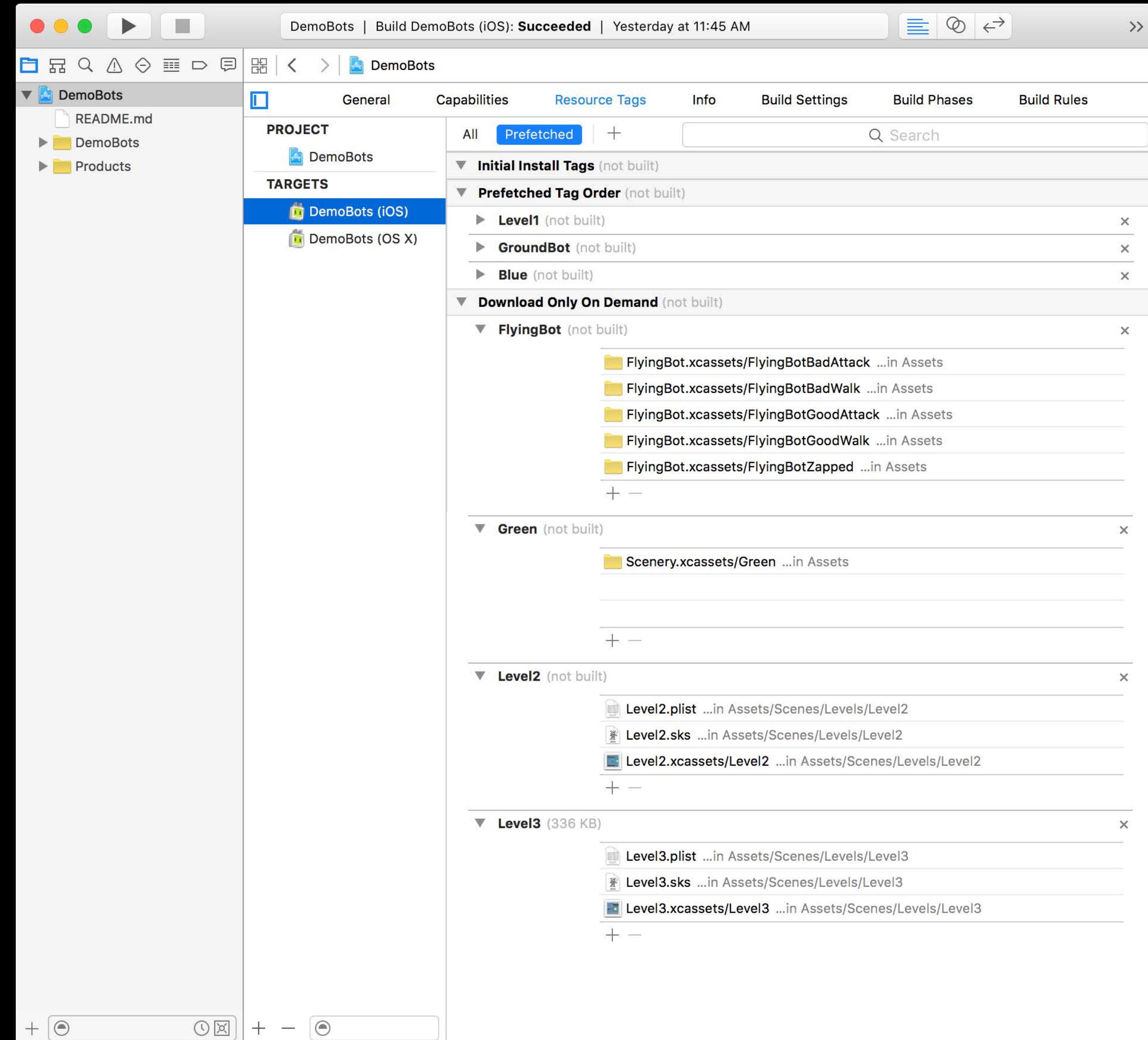
Level3

GroundBot



FlyingBot





▼ **Prefetched Tag Order** (not built)

- ▶ **Level1** (not built) x
- ▶ **GroundBot** (not built) x
- ▶ **Blue** (not built) x

▼ **Download Only On Demand** (not built)

- ▼ **FlyingBot** (not built) x
 - 📁 FlyingBot.xcassets/FlyingBotBadAttack ...in Assets
 - 📁 FlyingBot.xcassets/FlyingBotBadWalk ...in Assets
 - 📁 FlyingBot.xcassets/FlyingBotGoodAttack ...in Assets
 - 📁 FlyingBot.xcassets/FlyingBotGoodWalk ...in Assets
 - 📁 FlyingBot.xcassets/FlyingBotZapped ...in Assets

+ -

On Demand Resources

In DemoBots

On Demand Resources

In DemoBots

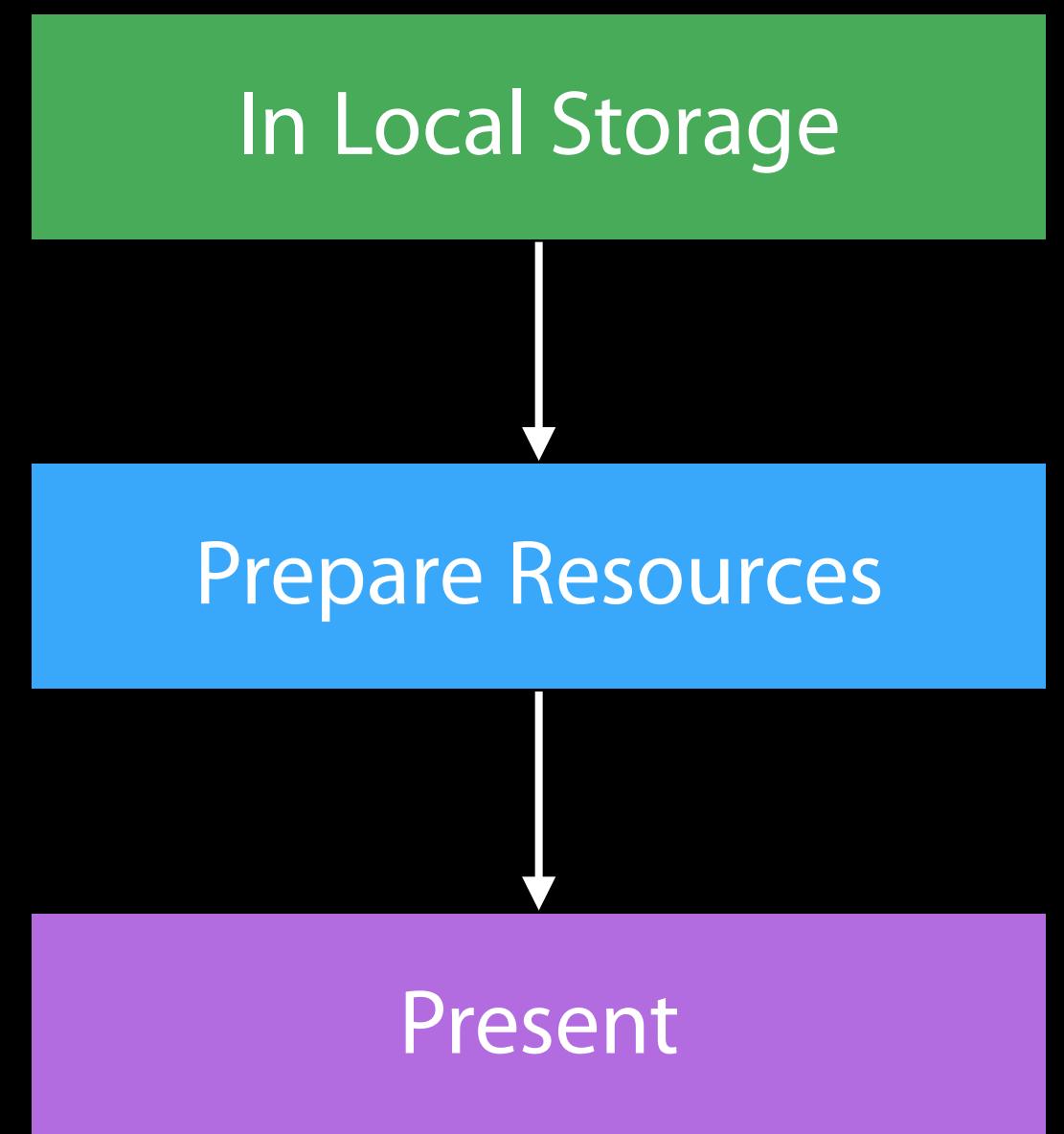
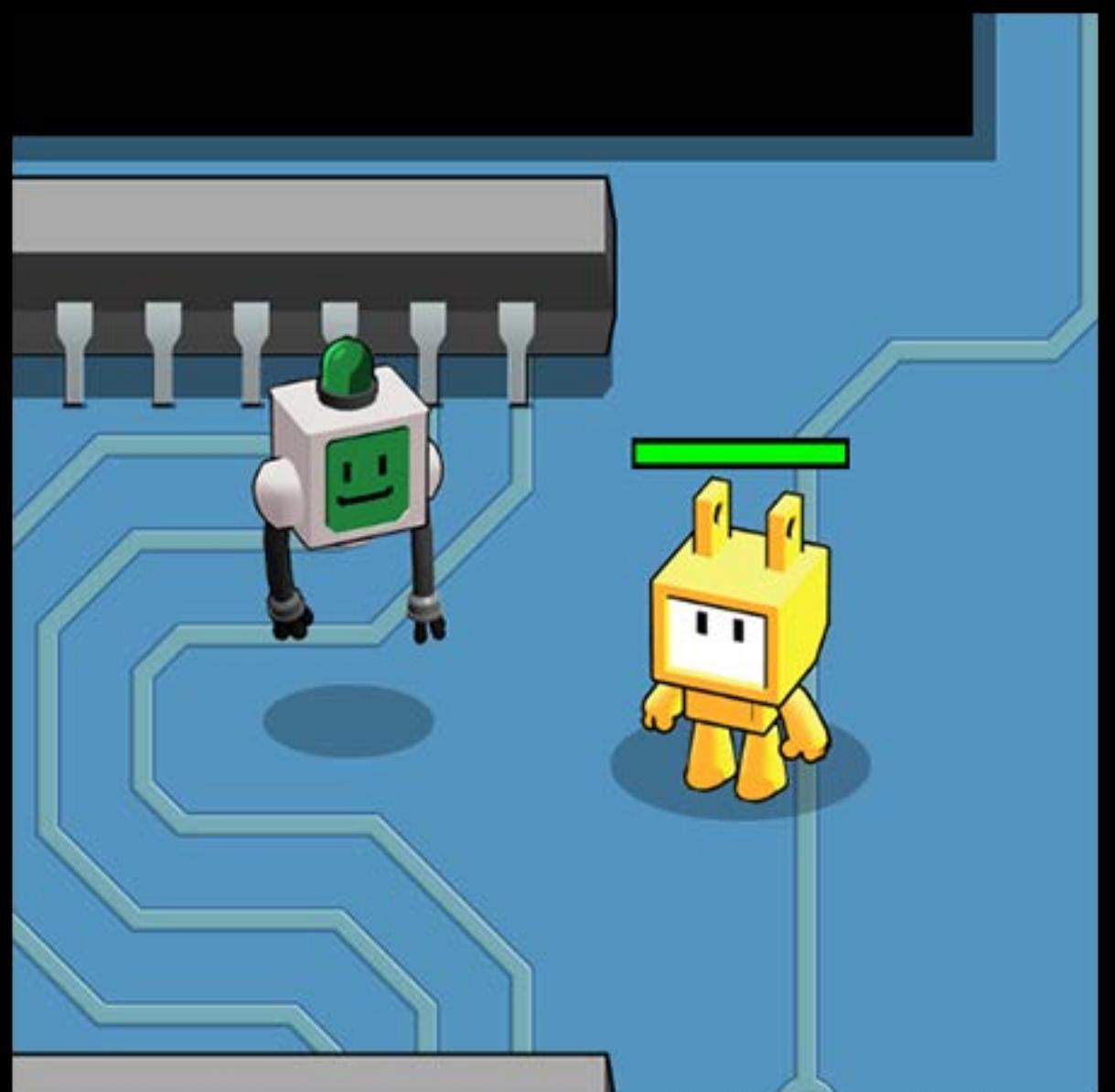
Tag resources for later download

- Faster initial download time
- Keep overall storage footprint small

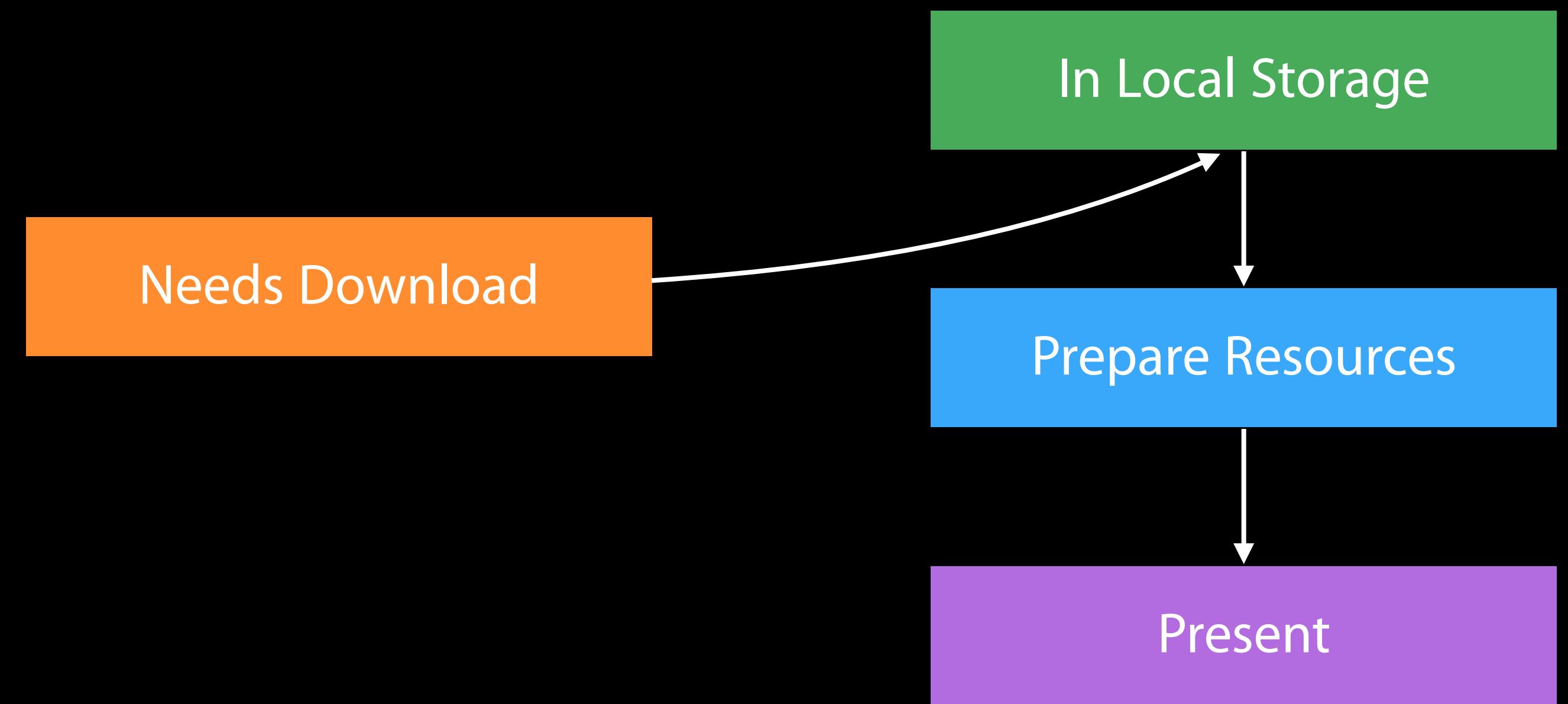
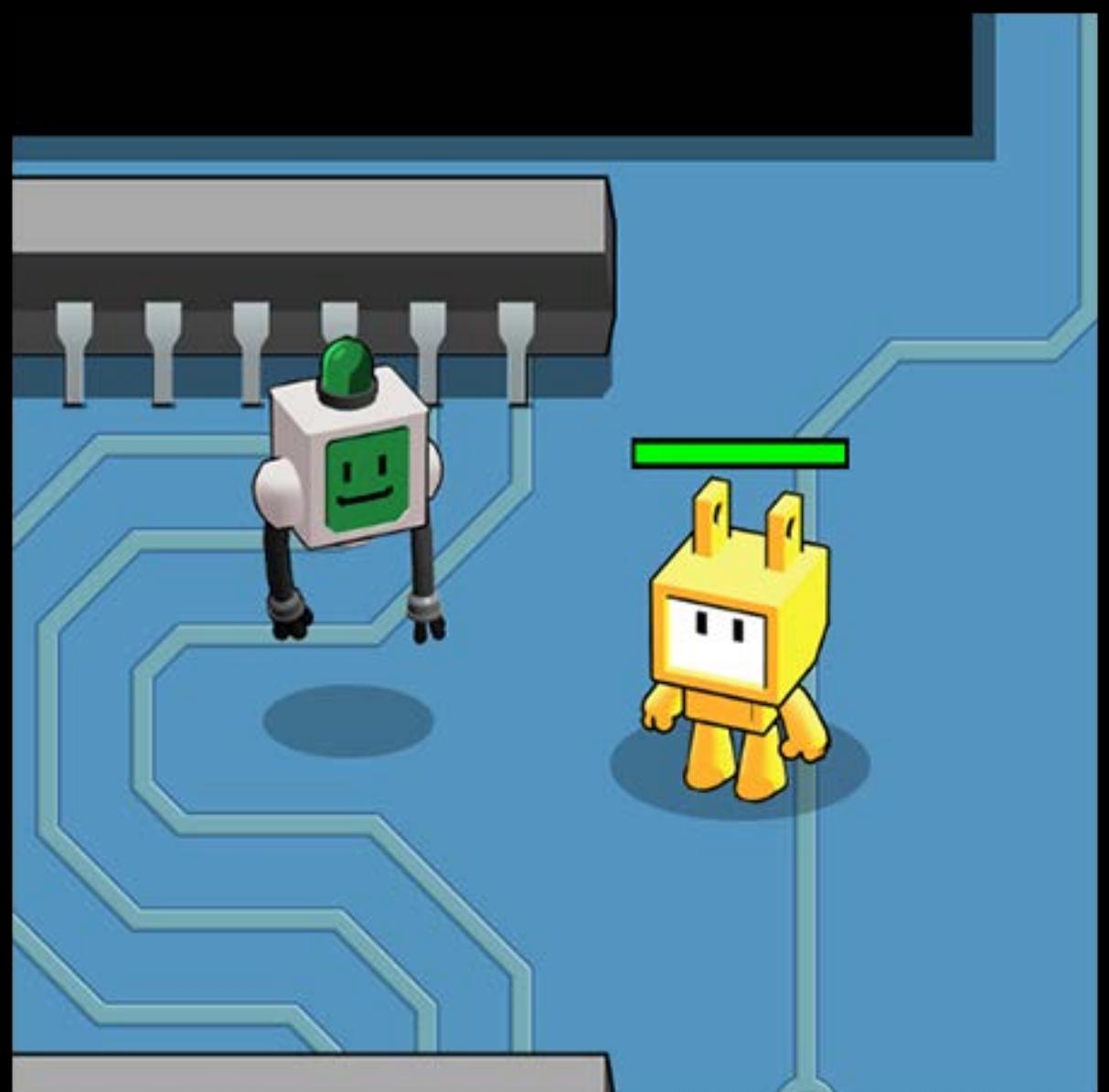
Present Next Scene



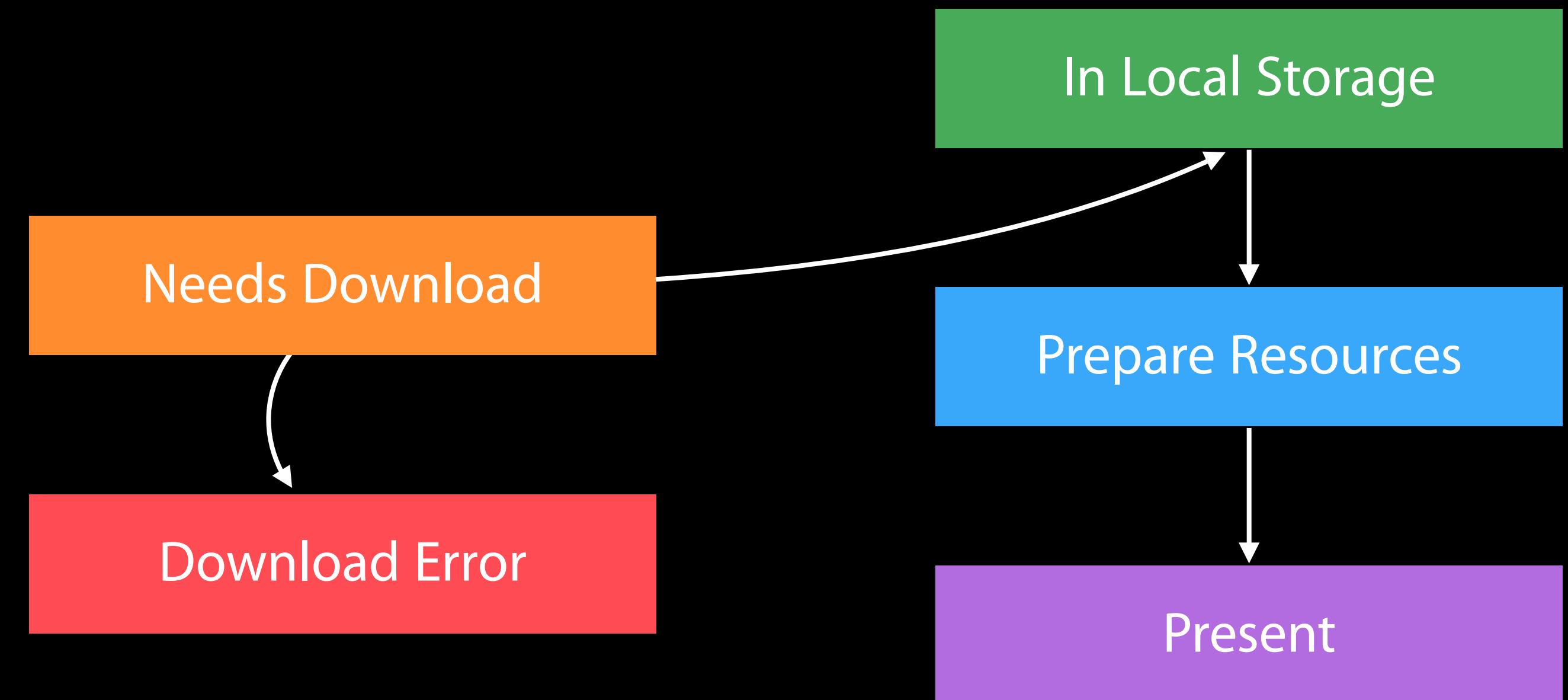
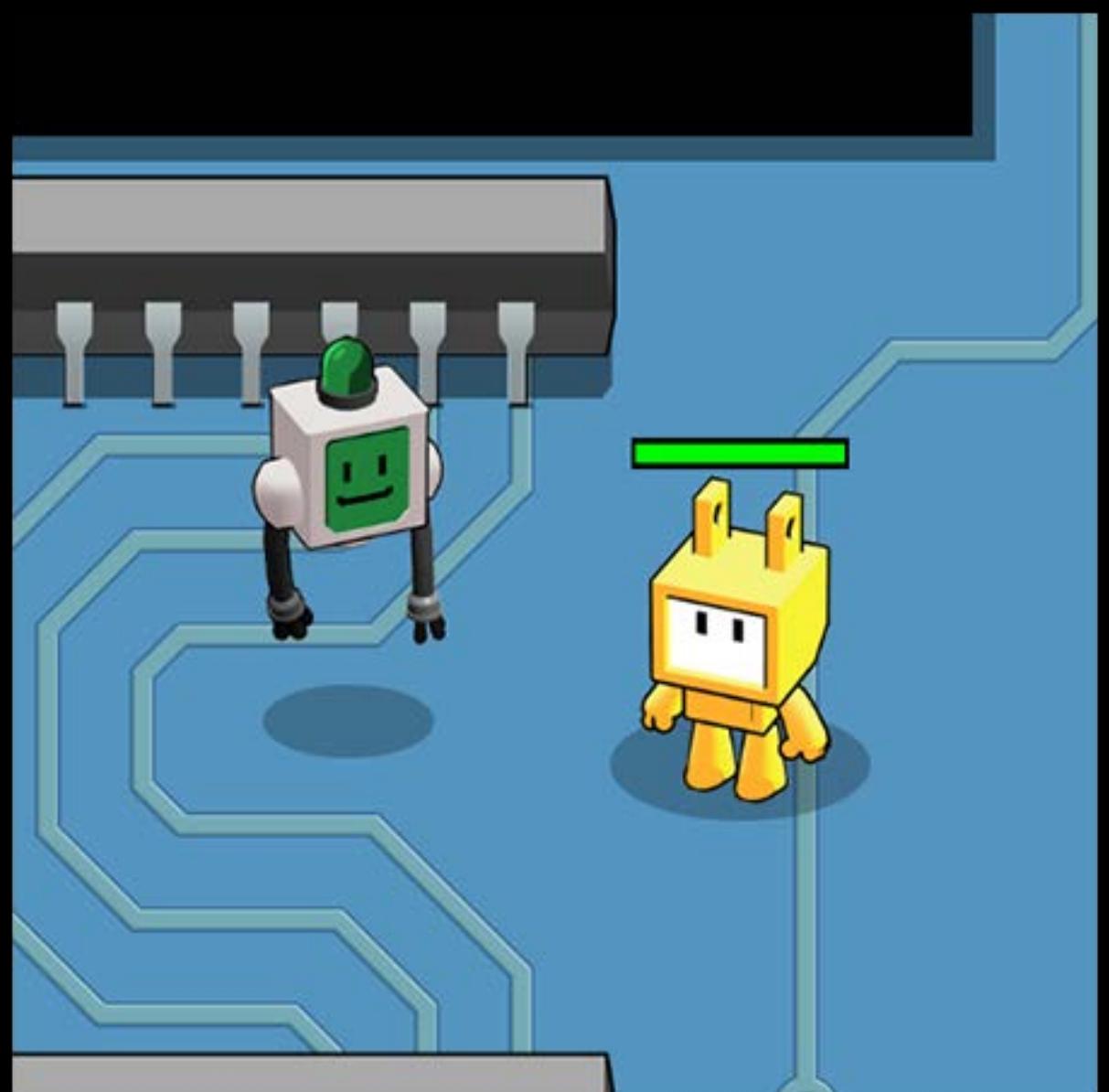
Present Next Scene



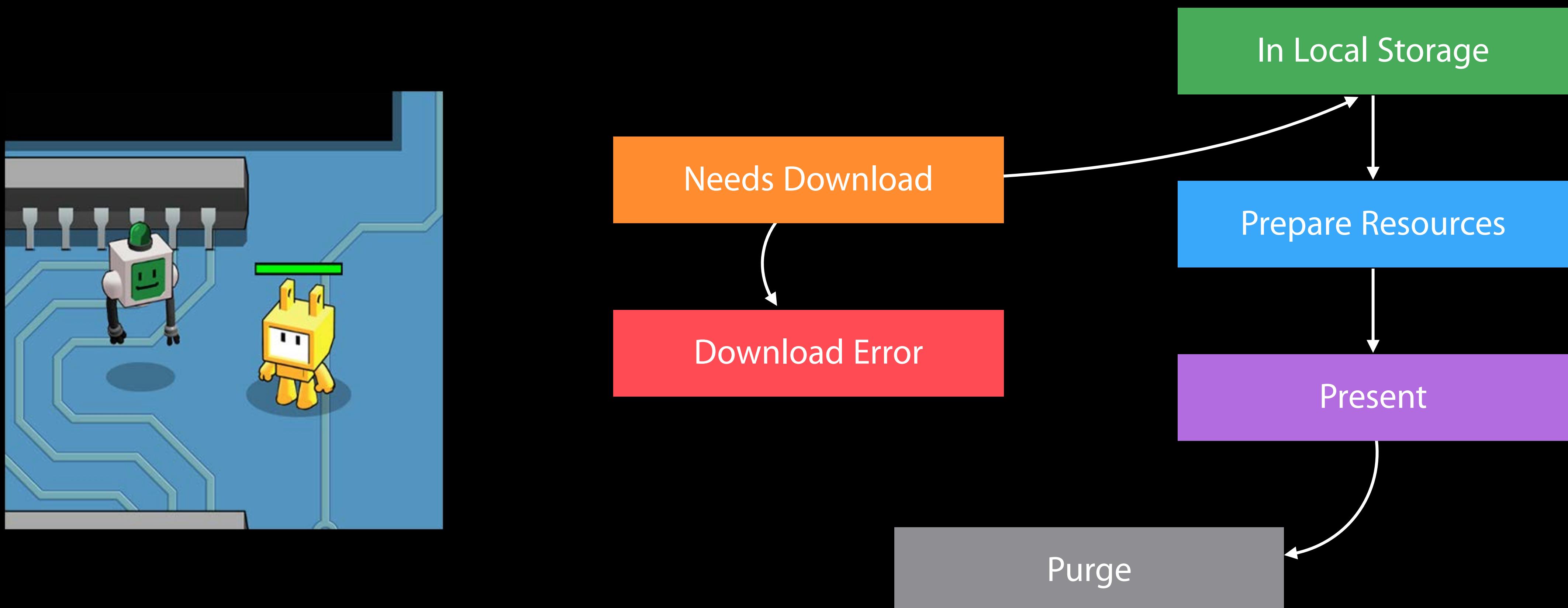
Present Next Scene



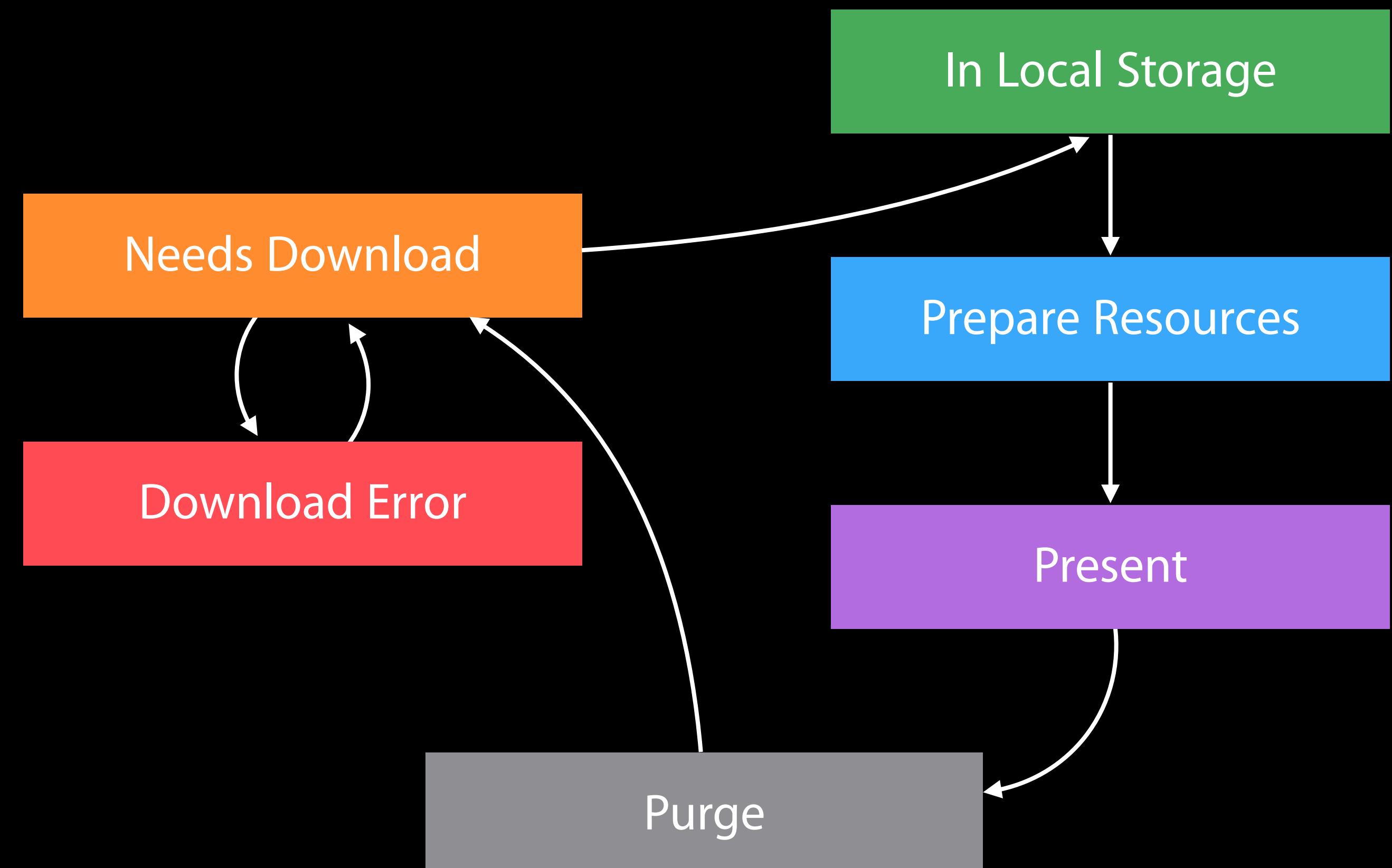
Present Next Scene



Present Next Scene



Present Next Scene



SceneLoader

SceneLoader

Initial

Downloading Resources

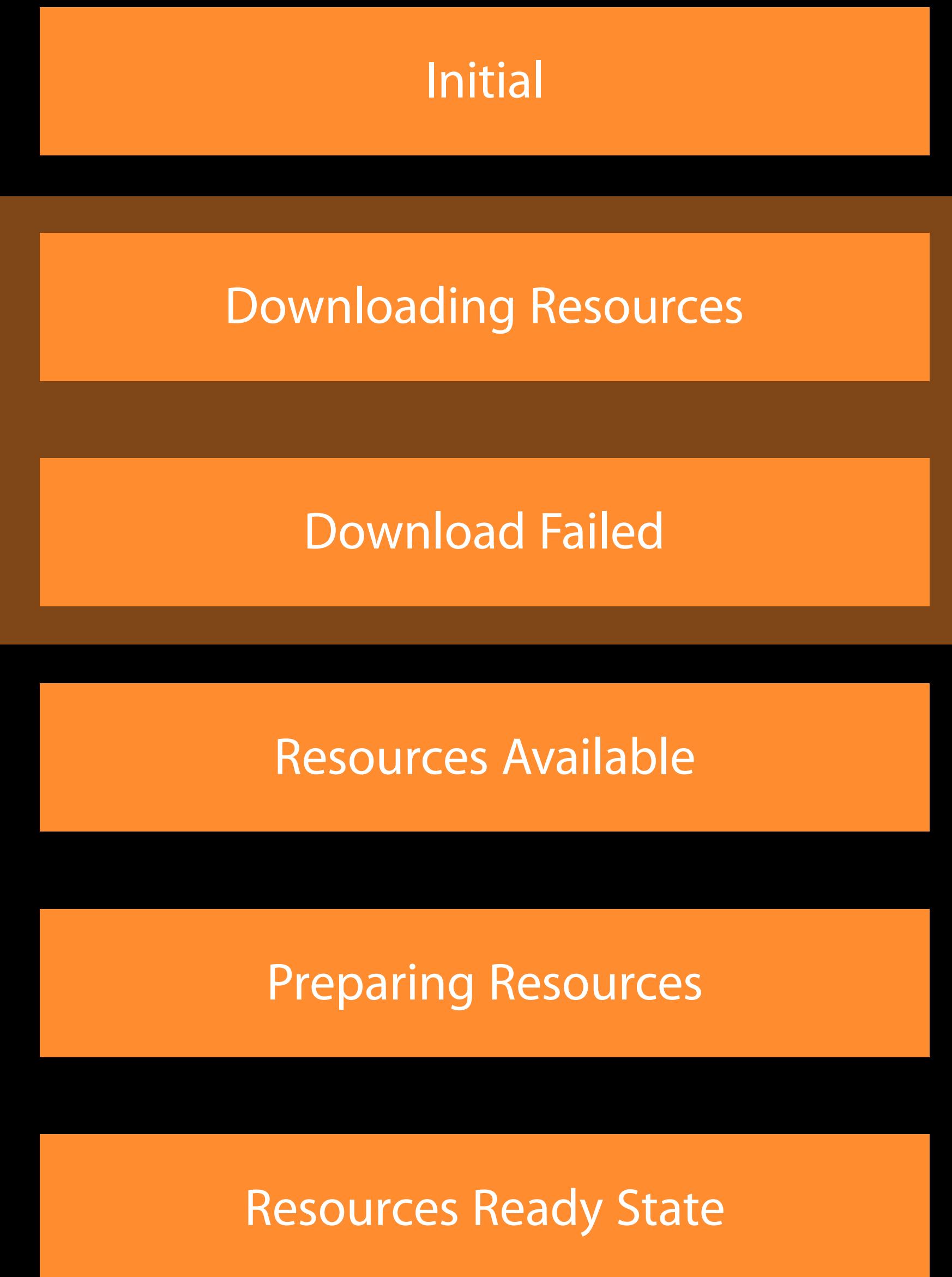
Download Failed

Resources Available

Preparing Resources

Resources Ready State

SceneLoader



SceneLoader

Preparing Resources

SceneLoader

Preparing Resources

```
override func isValidNextState(stateClass: AnyClass) -> Bool {
```

```
}
```

SceneLoader

Preparing Resources

```
override func isValidNextState(stateClass: AnyClass) -> Bool {  
    switch stateClass {  
        case is SceneLoaderResourcesReadyState.Type  
            where sceneLoader.scene != nil:  
                return true  
  
        case is SceneLoaderResourcesAvailableState.Type:  
            return true  
  
        default:  
            return false  
    }  
}
```

Final Tips

Final Tips

Start downloading resources early

- Request **Level2** as soon as **Level1** begins

Disk Report

DemoBots PID 965

CPU 12%
Memory 97.6 MB
Energy Impact High
Disk 8 KB/s
Network Zero KB/s
FPS

Disk

Profile in Instruments

Reading Writing

0.0 KB/s 0.0 KB/s
Per Second Per Second

6.5 MB 0.0 KB
Total Total

Reading and Writing Rates

372.0 KB/s
0.0 KB/s

293 352

■ Read ■ Written

On Demand Resources

Tag	Size	Status	Action
Green	143 KB	Downloaded	Purge
Level1	229 KB	In Use	Purge
FlyingBot	1.7 MB	Downloaded	Purge
Blue	156 KB	In Use	Purge
GroundBot	2.1 MB	In Use	Purge
Level2	356 KB	Downloaded	Purge
Level3	373 KB	Purged	

Open Files

Descriptor	Type	Device	Size/Offset	Inode	Path
Or	CHR	50,331,650	0	100	/dev/null

Demobots

On Demand Resources

Tag	Size	Status	Actions
Green	143 KB	Downloaded	Purge
Level1	229 KB	In Use	
FlyingBot	1.7 MB	Downloaded	Purge
Blue	156 KB	In Use	
GroundBot	2.1 MB	In Use	
Level2	356 KB	Downloaded	Purge
Level3	373 KB	Purged	

Final Tips

Start downloading resources early

- Request **Level2** as soon as **Level1** begins

Final Tips

Start downloading resources early

- Request **Level2** as soon as **Level1** begins

Modify the priority as needed

```
bundleResourceRequest.loadingPriority = 0.8  
operationQueue.qualityOfService = .UserInitiated
```

DemoBots

DemoBots

Wanted

Reduce number of animation frames

Fine tune assets for each device

Constrain camera to player and level

Enforce character state transitions

Share functionality between characters

Complex situational decision making

Elegant character navigation

Smooth natural movement

Add assets to improve gameplay

Speed up initial download

Model resource loading

DemoBots

Wanted

Reduce number of animation frames

Fine tune assets for each device

Constrain camera to player and level

Enforce character state transitions

Share functionality between characters

Complex situational decision making

Elegant character navigation

Smooth natural movement

Add assets to improve gameplay

Speed up initial download

Model resource loading

Technology

SpriteKit actions editor

Texture atlases in asset catalogs

SKCameraNode

GKStateMachine

GKEntity and **GKComponent**

Fuzzy Logic with **GKRuleSystem**

Pathfinding via **GKObstacleGraph**

GKAgent2D, **GKBehavior**, and **GKGoal**

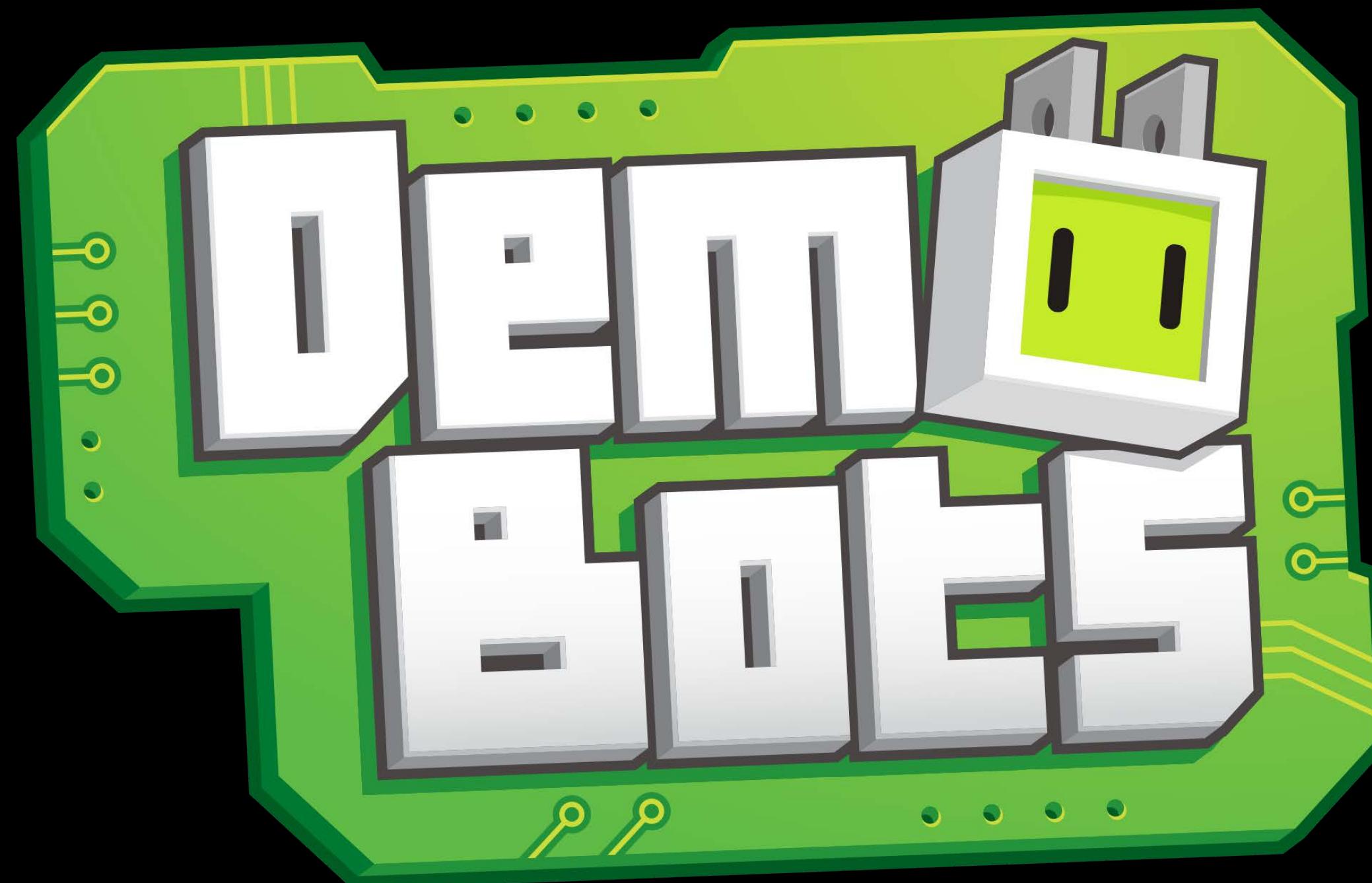
Asset catalogs with App Slicing

On Demand Resources

GKStateMachine

Download Sample Files

<http://developer.apple.com/spritekit>



More Information

Documentation and Videos

<http://developer.apple.com>

Apple Developer Forums

<http://developer.apple.com/forums>

Developer Technical Support

<http://developer.apple.com/support/technical>

General Inquiries

Allan Schaffer, Game Technologies Evangelist

aschaffer@apple.com

Related Sessions

App Thinning in Xcode	Presidio	Wednesday 9:00AM
What's New in SpriteKit	Mission	Wednesday 10:00AM
Going Social with ReplayKit and Game Center	Mission	Wednesday 1:30PM
Introducing On Demand Resources	Pacific Heights	Wednesday 4:30PM
Introducing GameplayKit	Mission	Thursday 11:00AM

Labs

GameplayKit Lab

Graphics, Games,
and Media Lab C

Thursday 2:30PM

SpriteKit Lab

Graphics, Games,
and Media Lab C

Friday 9:00AM

GameplayKit Lab

Graphics, Games,
and Media Lab C

Friday 12:00PM

 **WWDC 15**