



KubeCon



CloudNativeCon

North America 2024





KubeCon

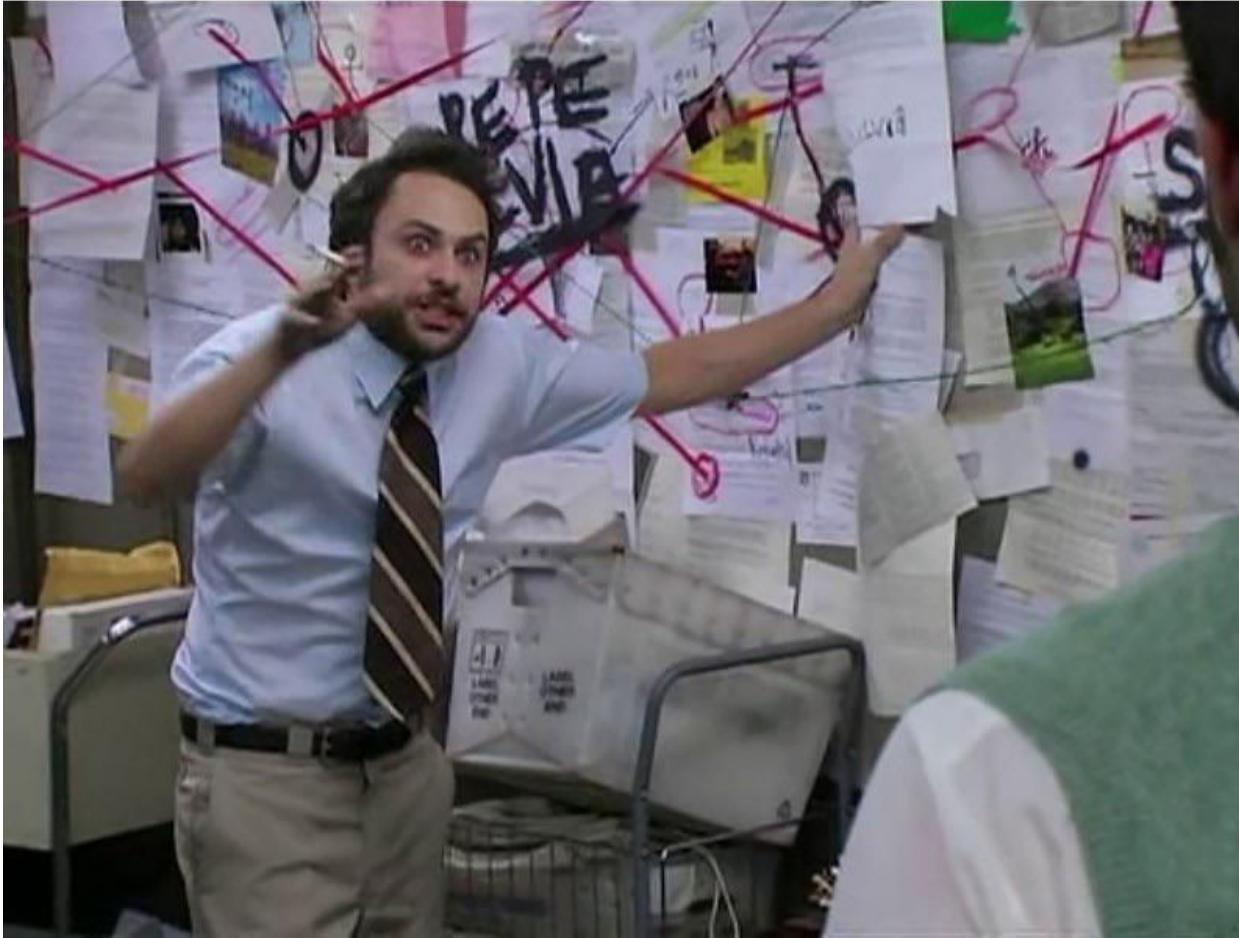


CloudNativeCon

North America 2024

# Scratching the Surface: Simulating K8s in MIT Scratch

*Jude Connors & Mitch Connors*





KubeCon



CloudNativeCon

North America 2024



**Mitch Connors**  
Software Engineer  
*Microsoft*



**Jude Connors**  
Professional  
Non-expert

# What is Kubernetes? ELI14



*ReLogo ai*

# Core Concept: Declarative Interface





# What is Scratch?

The image shows the Scratch programming interface. At the top, there's a purple header bar with the word "SCRATCH" in yellow, followed by "Settings", "File", "Edit", "Tutorials", "Join Scratch", and "Sign in". Below the header is a toolbar with icons for "Code", "Costumes", and "Sounds". On the left, a "Code" palette lists categories: Motion (move 10 steps, turn 15 degrees, turn 15 degrees), Looks, Sound, Events, Control (go to random position), Sensing (go to x: 0 y: 0), Operators (glide 1 secs to random position, glide 1 secs to x: 0 y: 0), Variables, and My Blocks (point in direction 90, point towards mouse-pointer). The main area is the stage, which features a grid and a yellow cat sprite running towards the right. To the right of the stage is the "Sprite" palette, which shows "Sprite1" with a size of 100 and a direction of 90. Below the stage is the "Stage" palette, which includes sections for "Backdrops" (with 1 backdrop) and "Backgrounds" (with 1 background).



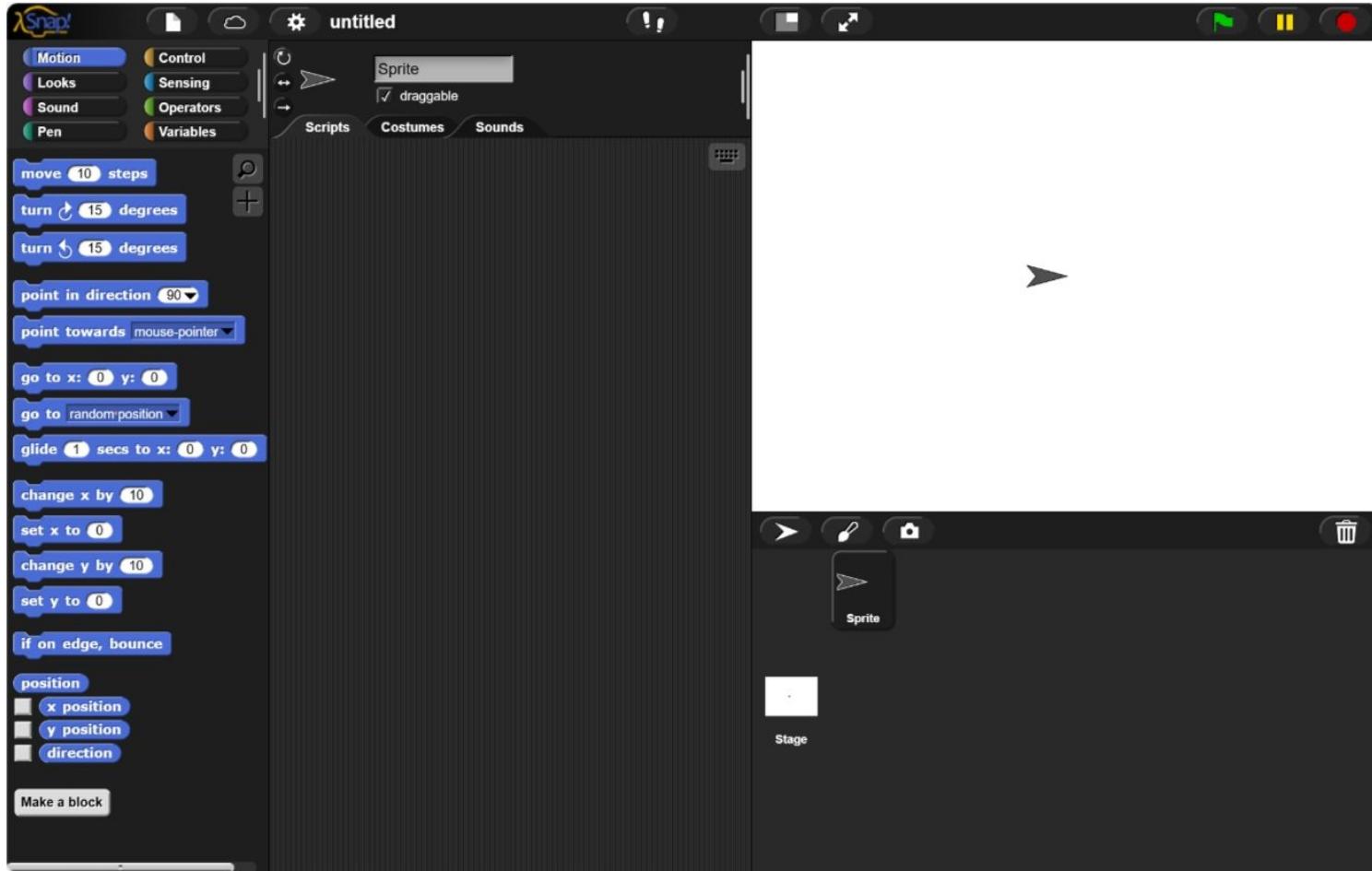
KubeCon



CloudNativeCon

North America 2024

# What is Snap?





KubeCon



CloudNativeCon

North America 2024

# Agenda

- Module 1 - The Scheduler
- Module 2 - The Replicator
- Module 3 - The Deployment

## What it isn't

- Not Running Containers
- No REST Server
- No YAML
- Lack of strong typing
- Snap not Scratch
- No Status

## What it is

- Declarative Object Model
- Prototyping
- Simulated Demonstration

# A Brief Tour of the Codebase

New Node

New Pod

Get Node

Get Pods

Get Size

Get Name

Is > a Node

Is > a Pod

All Nodes

All Pods

Add Pod to me >

Set my Node >

node capacity

debug node

node placer

Write Text

My lowest pod bottom

Pick Node for >

New ReplicaSet

All ReplicaSets

Is > a ReplicaSet

Get ReplicaSet

schedulePod

hasdf

Add Pod to RS >

Remove Pod from me >

New Deployment

Is > a Deployment

Get Replicas

Update RS replicas

Set Deployment Size

# Module 1 - The Scheduler

## Vocabulary:

- Node
- Pod
- Immutable
- Kubelet

## Exceptions:

- Pod reduced to Name and Size
- No difference between CPU, Memory, Requests vs Limits
- No Containers

## The Scheduler's Job:

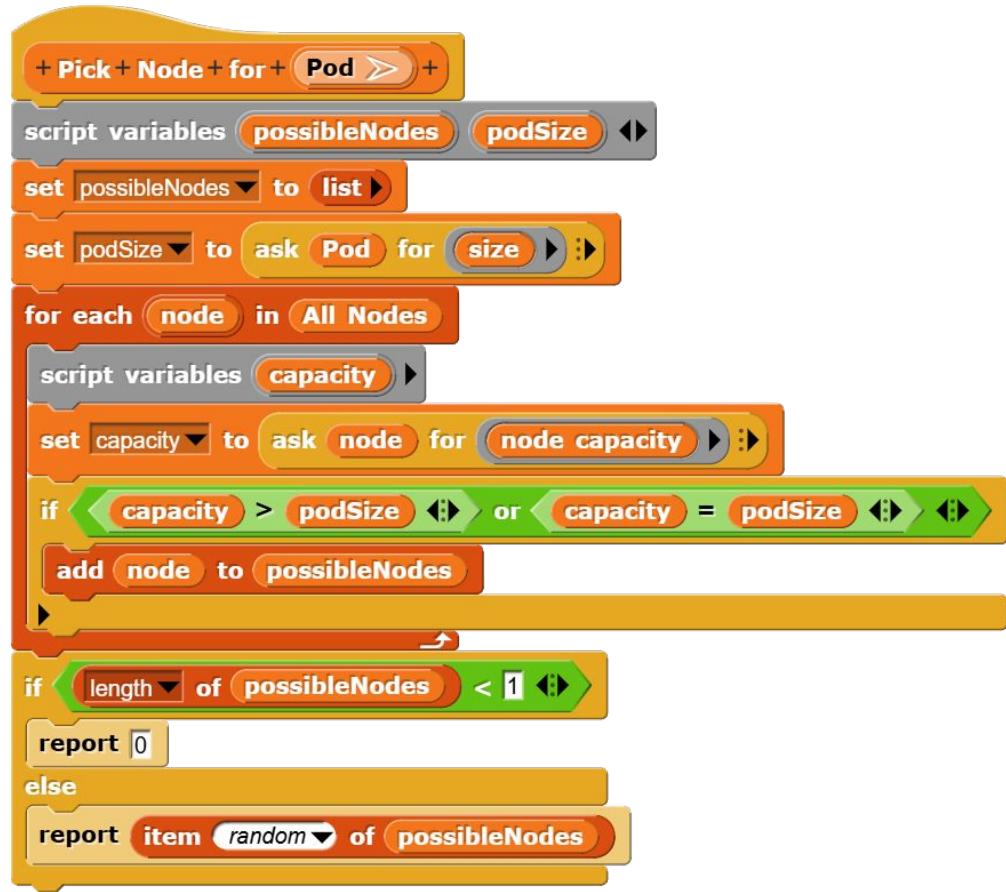
- Make sure all pods are assigned to nodes that have space for them

# Module 1 - The Scheduler

```
+ schedulePod + newName + newSize +  
script variables pod node yPos ◀▶  
set pod ▾ to New Pod newName newSize  
set node ▾ to Pick Node for pod  
set yPos ▾ to ask node for My lowest pod bottom ▶▶;  
if node = 0 ◀▶  
tell pod to delete this clone ▶▶;  
say This pod is unschedulable for 2 secs  
else if ✓
```

```
else if ✓  
tell pod to Set my Node node ▶▶;  
tell node to Add Pod to me pod debug node ▶▶;  
tell pod to  
go to node  
change x by 4.5  
set y to yPos  
Write Text 30 + ask node for my left ▶▶ ▶▶ yPos  
newName  
report pod
```

# Module 1 - The Scheduler



# Module 1 - The Scheduler



# Module 2 - The Replicator

## Vocabulary:

- ReplicaSet
- Replicas

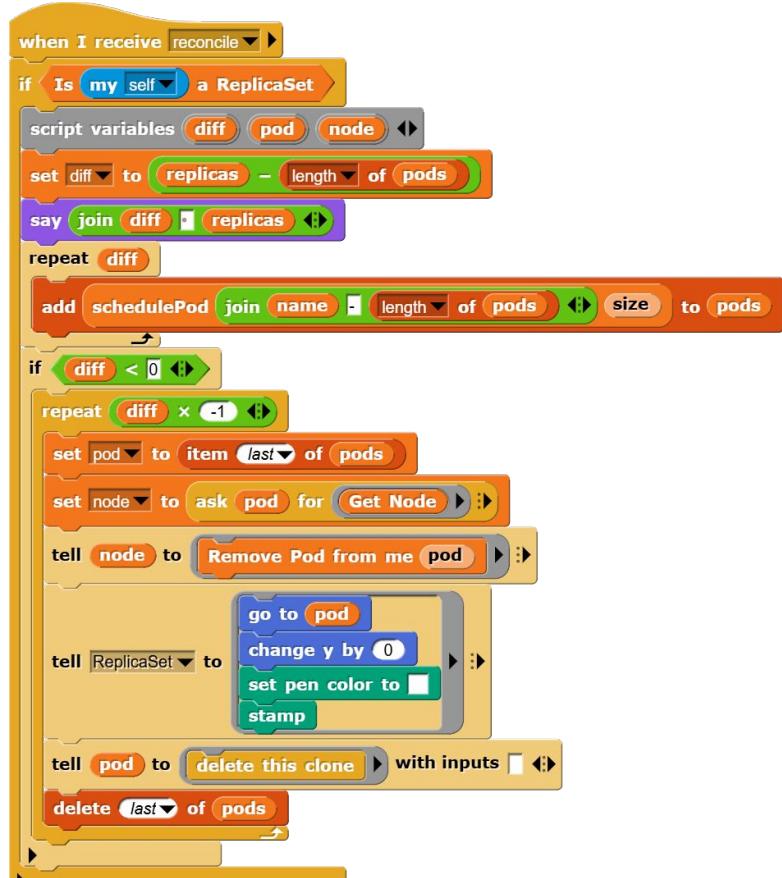
## The Replicator's Job:

Ensure there are always exactly as many pods as replicas running

## Exceptions:

- No way to delete, modify ReplicaSets in UI
- Actually called the Replication Controller, but Replicator sounds cooler
- Can't watch individual objects, all objects reconcile on broadcast

# Module 2 - The Replicator



# Module 2 - The Replicator



# Module 3 - The Deployment

## Vocabulary:

- Deployment
- Progressive Rollout

## The Deployment's Job:

Transition from one version to another in a progressive, controlled manner.

## Exceptions:

- No pod readiness
- No Kubectl



# Module 3 - The Deployment

```
when I receive reconcile
script variables newReplicas oldReplicas currentReplicas
if Is my self a Deployment
if newRS = 0
set newRS to New ReplicaSet join name - version size replicas
else
if not join name - version = ask newRS for Get Name
set oldRs to newRS
set newRS to New ReplicaSet join name - version size 1
set newReplicas to ask newRS for Get Replicas
set oldReplicas to ask oldRs for Get Replicas
set currentReplicas to newReplicas + oldReplicas
```

```
if replicas > currentReplicas
tell newRS to Update RS replicas replicas - oldReplicas
else if currentReplicas > replicas and oldReplicas > 0
set oldReplicas to 0 max oldReplicas - currentReplicas - replicas
tell oldRs to Update RS replicas oldReplicas
set currentReplicas to newReplicas + oldReplicas
else if currentReplicas > replicas
tell newRS to Update RS replicas replicas
if 1 > oldReplicas
tell oldRs to delete this clone
set oldRs to 0
else if 
tell newRS to Update RS replicas newReplicas + 1
```

# Module 3 - The Deployment



# Declarative Resiliency





KubeCon



CloudNativeCon

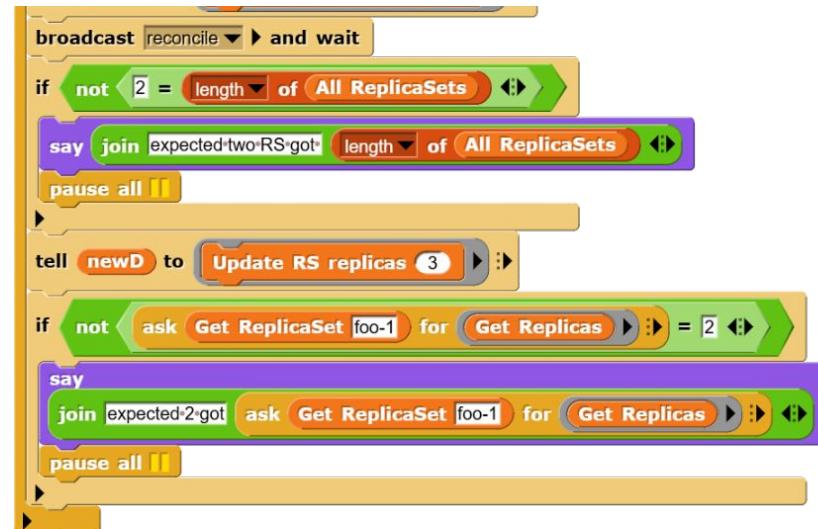
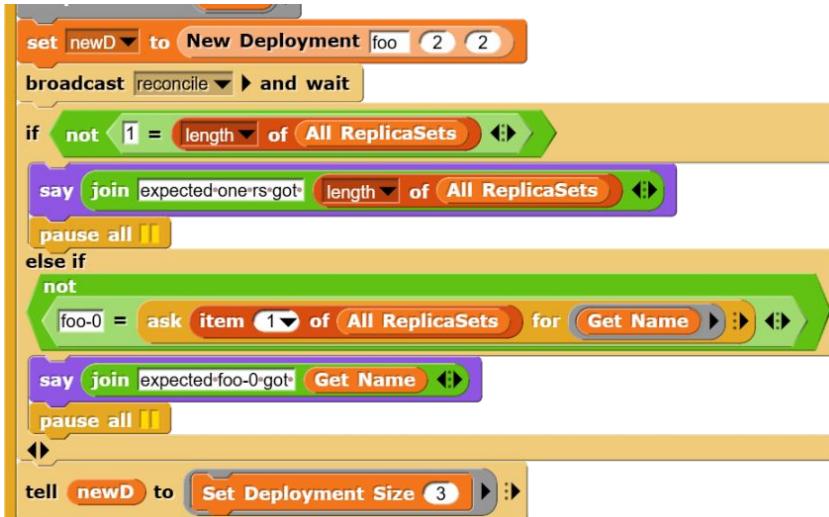
North America 2024

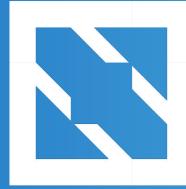
# What's Next?

The background features a dark blue gradient with a stylized mountain range silhouette at the bottom. Overlaid are several white snowflake icons of varying sizes and orientations, suggesting a winter or cold environment.

- Services
- Endpoints
- Gateways
- ConfigMaps

# Chopping Wood - Unit Tests





KubeCon

CloudNativeCon

North America 2024

Thank You!

