Playing Ping Pong on Kubernetes v4

A Study on Kubernetes Resilience

Eduardo Patrocinio, eduardop@us.ibm.com, @patrocinio
Sahdev Zala, spzala@us.ibm.com, @sp_zala

Objective

- Describe how to make an application resilient
- Validate Kubernetes resilience

Scenario

Two components (players) exchanging messages (balls) continuously

Goal

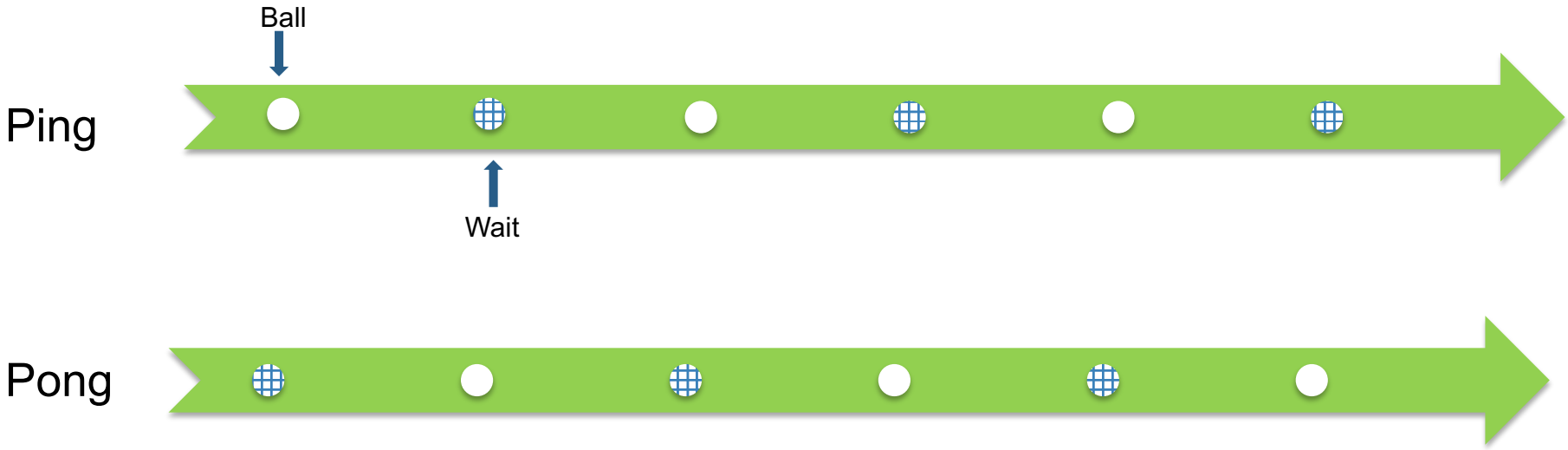
Keep the ball moving (despite some disruptions)

Rules of the Game

Each player

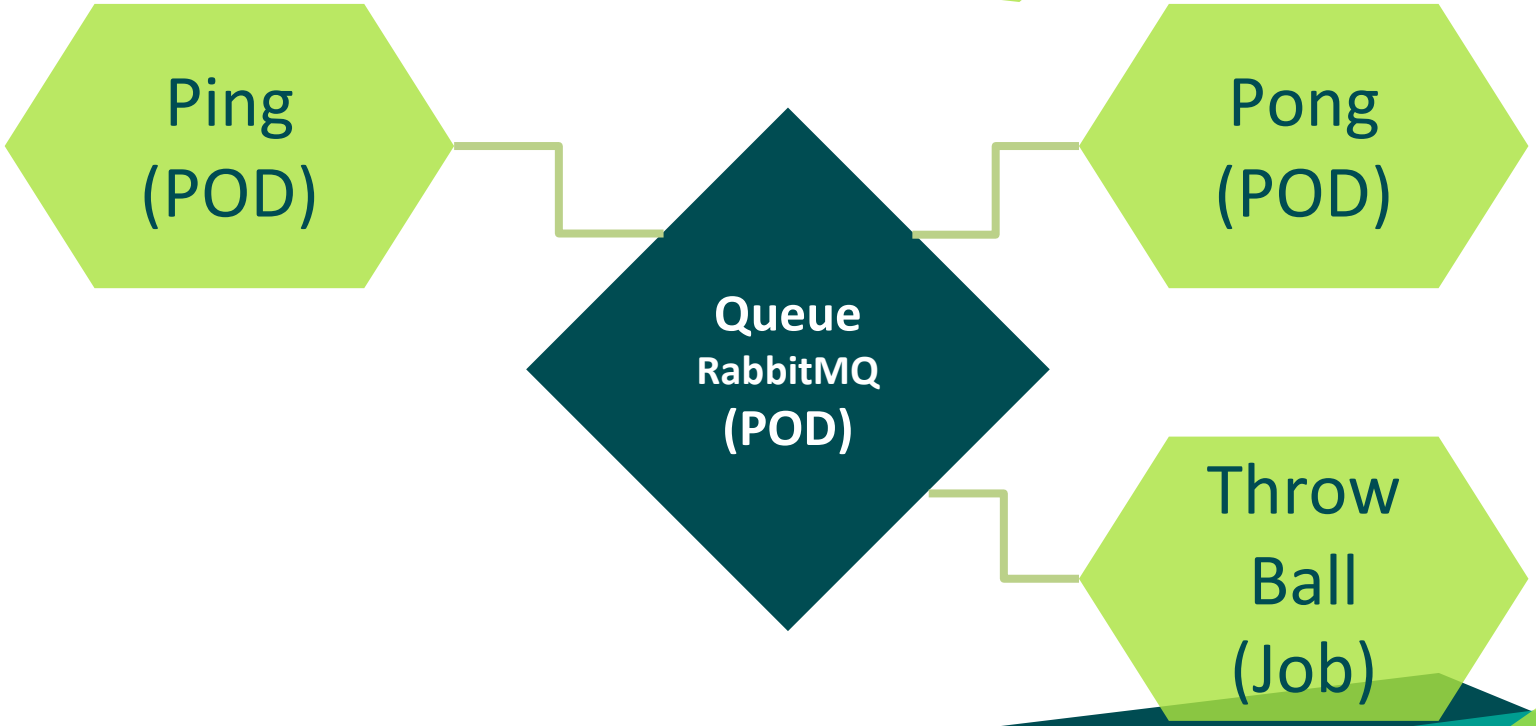
- Takes a ball (message) from a channel
- **Holds the ball for one second**
- Releases the ball to another channel

Desired State



1st
Game
Set up

Architecture



Queue Architecture

Play time

2nd
Game
Let’s kill a Pod

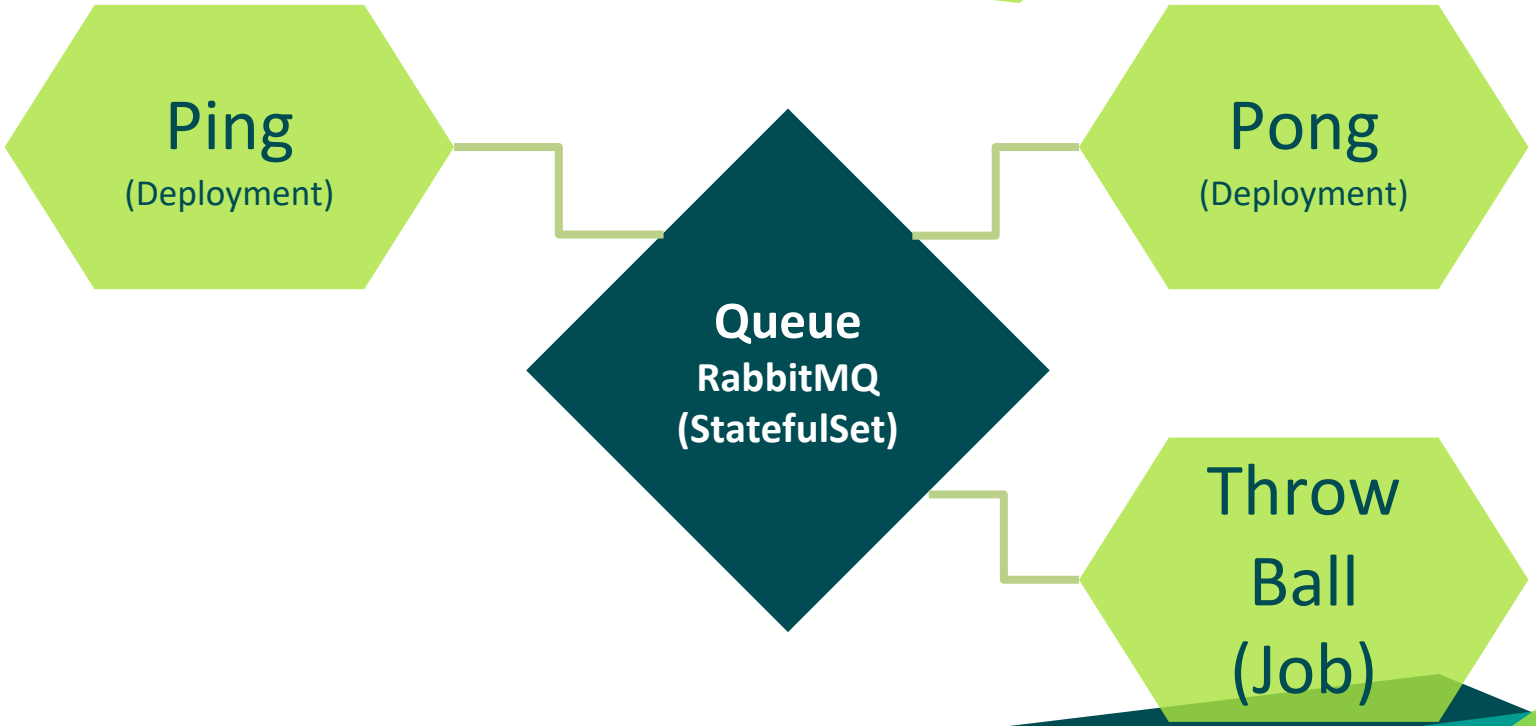
Task

- Kill the Ping Pod
- *What's going to happen?*

Play time

3rd
Game
Deployment

Architecture



Play time

4th
Game

Let’s kill the Pod in the ReplicaSet

Task

- Kill the Ping Pod in the Ping Replica Set
- *What's going to happen?*

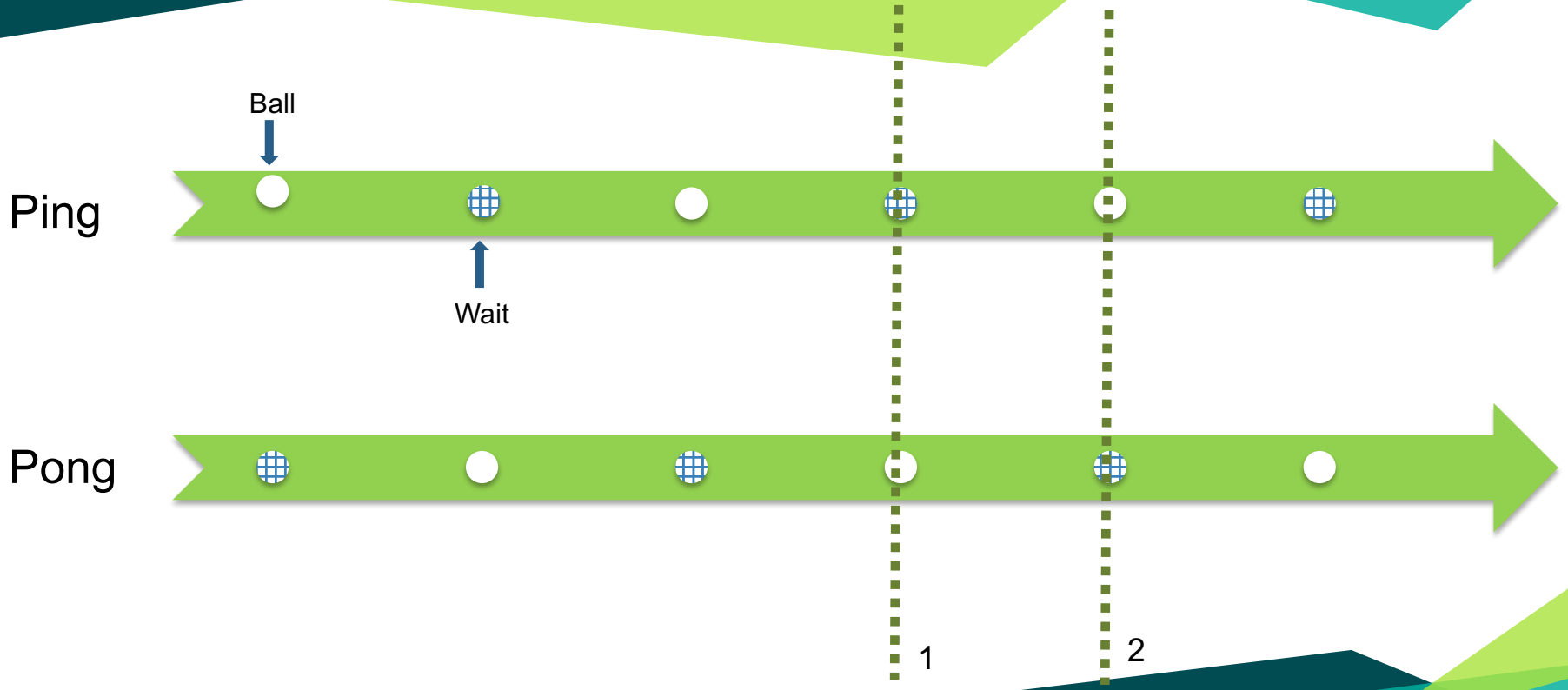
Play time

Logic Flow

Ping / Pong:

- Takes a ball (message) from a channel
- **Holds the ball for one second**
- Releases the ball to another channel

Timeline



5th
Game
Save the state

Logic Flow

Flow:

- Take a ball from a channel
- Store in the cache
- **Holds the ball for one second**
- Releases the ball to the other channel
- Clean the cache

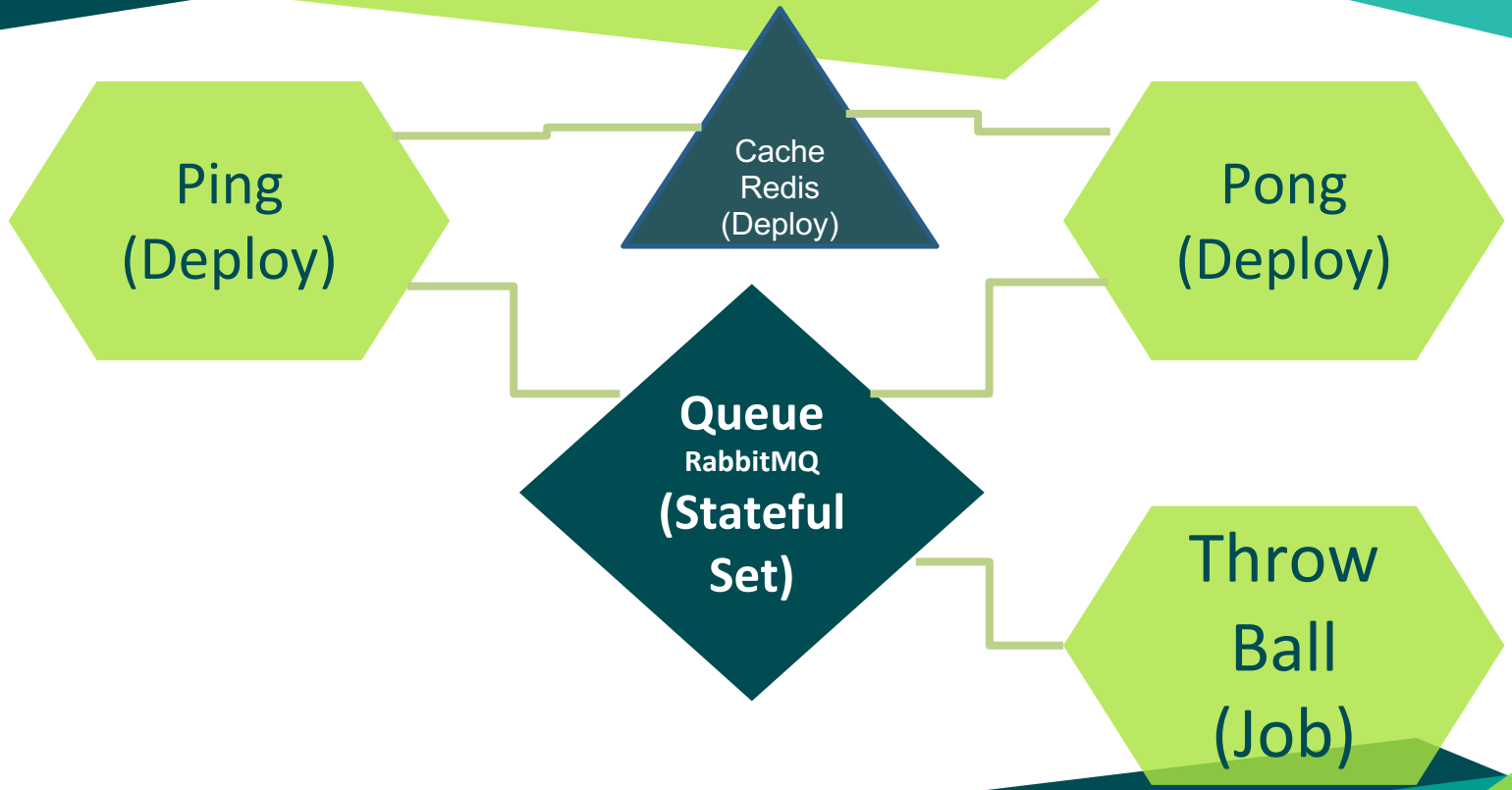
Initialization:

- Is there a ball in the cache?

  If so:

  - Releases the ball to the other channel
  - Clean the cache

Architecture



Play time

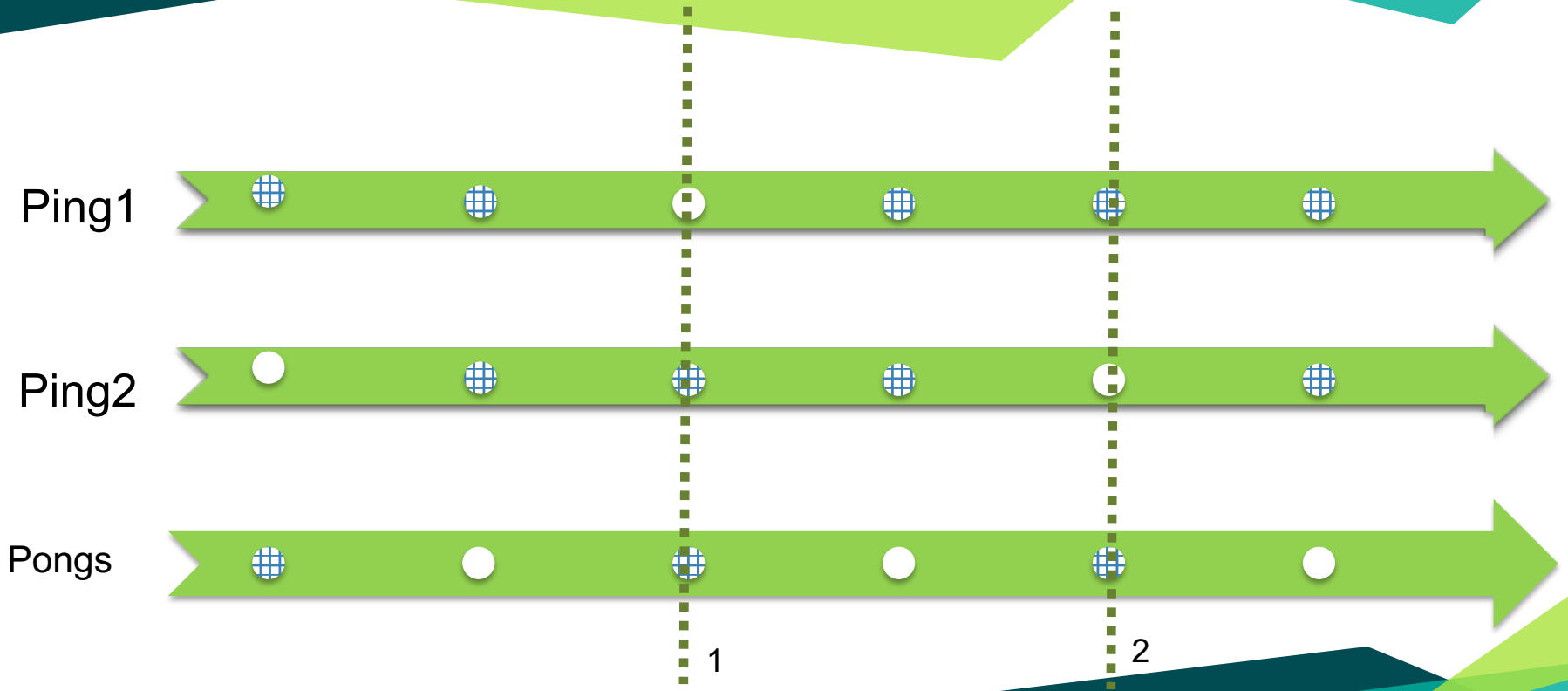
6th
Game
Doubles game

Task

- 2 Pings, 2 Pongs
- Kill the Ping Pod in the Replica Set
- *What's going to happen?*

Play time

Timeline



Scenario

- Ping2 has the ball
- Ping2 stores the ball in the cache
- I kill Ping1
- Kubernetes starts another Ping1
- During its initialization, Ping1 releases the ball
- Ping2 also releases the ball

7th
Game
StatefulSet

Logic Flow

Flow:

- Take a ball from a channel
- Store **(Pod, ball)** in the cache
- Holds the ball for one second
- Releases the ball to the other channel
- Clean the cache

Initialization:

- Is there a ball in the cache **for current Pod**?

  If so:

- Releases the ball to the other channel
- Clean the cache

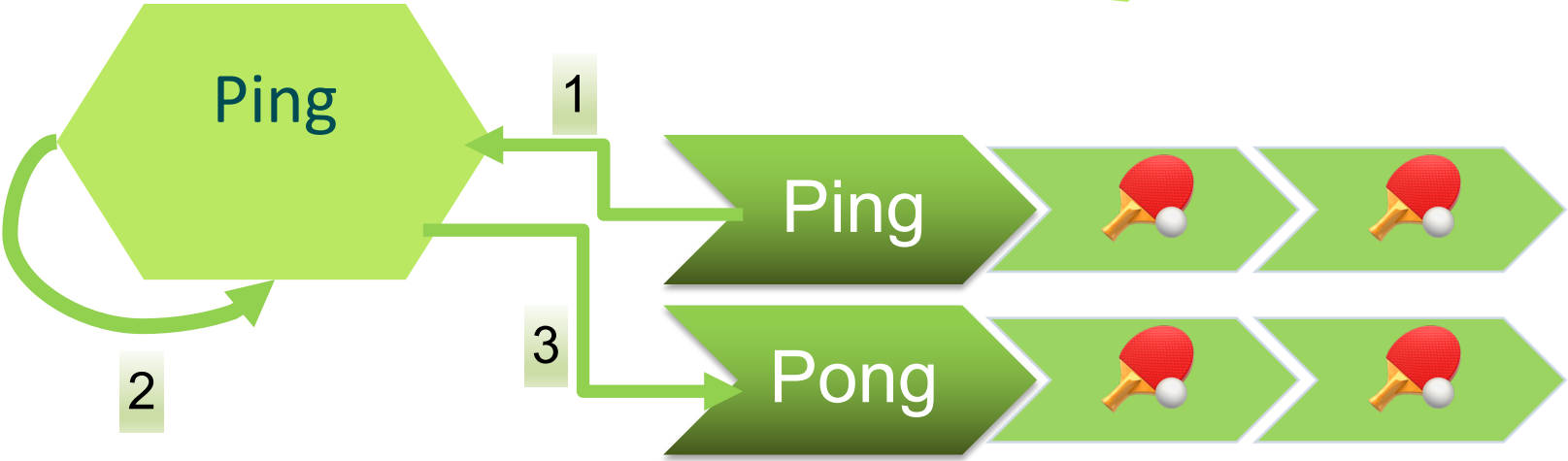
Play time

8th

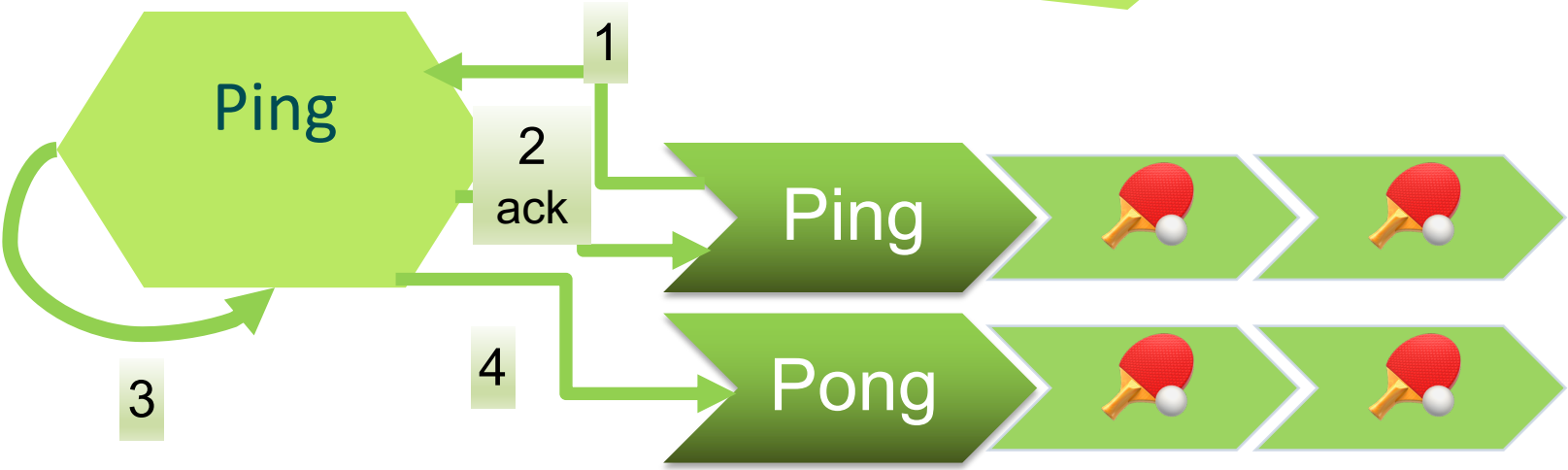
Game

A better (utilization of) Queue

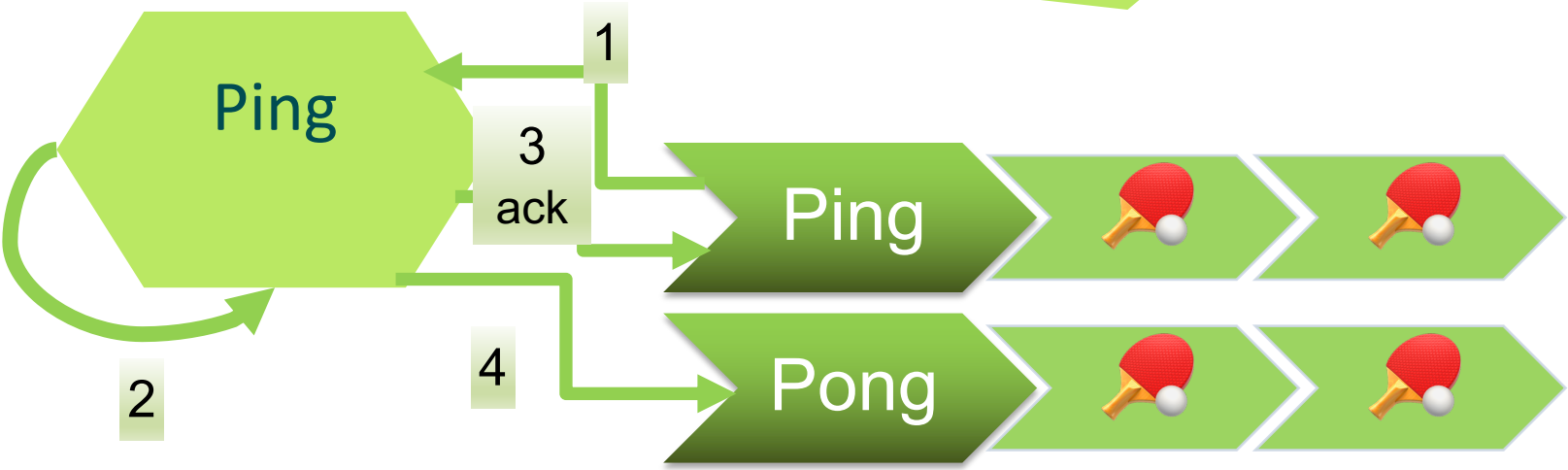
Current Queue Architecture



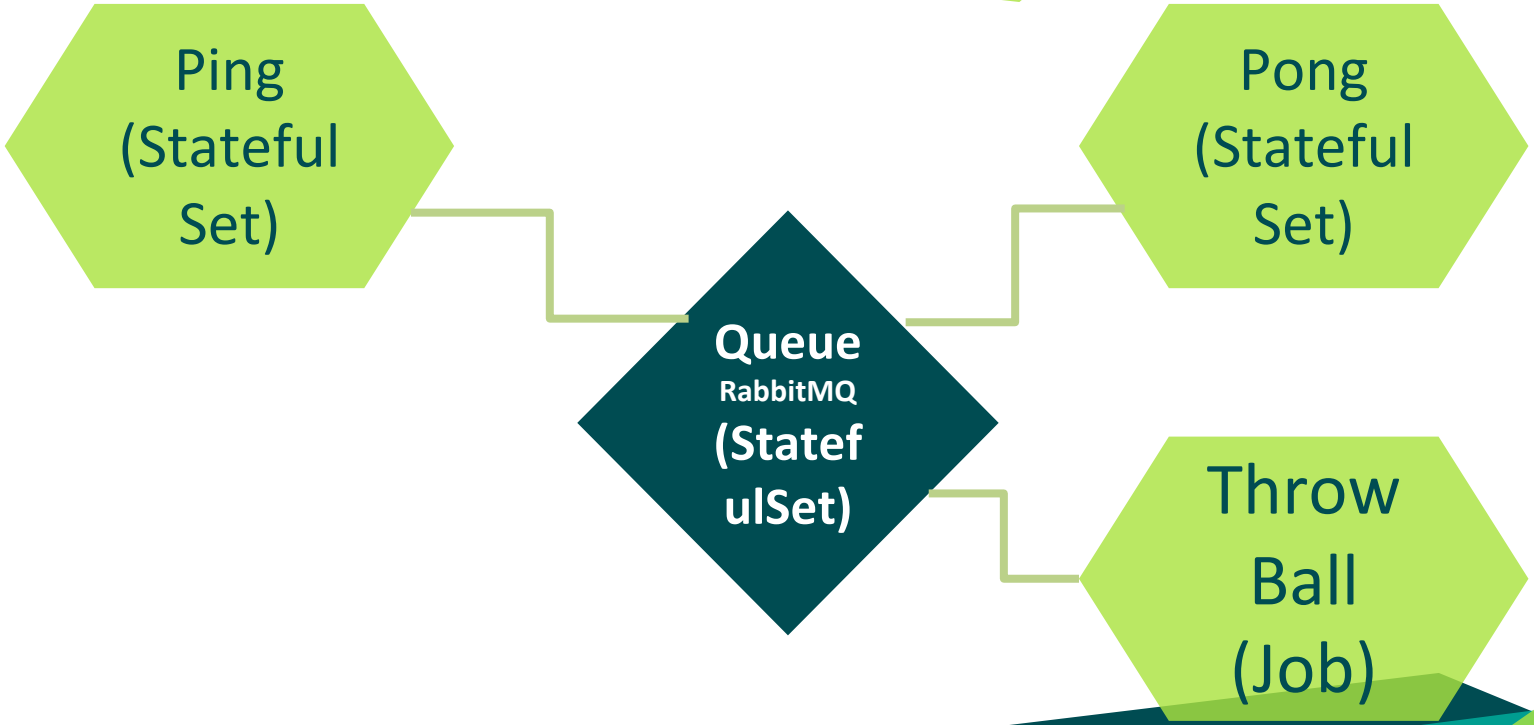
In fact, It's More Like This



So let's change the order…



Architecture



Play time

9th
Game
Time for platform disruption

Task

- Stop a Kubernetes node
- *What's going to happen?*

Play time

Conclusion

How can we achieve application resilience

- Application written with resilience in mind
- Resilience provided by Kubernetes

Thanks!

Any questions?

You can find us at @patrocinio or @sp_zala

https://github.ibm.com/eduardop/chaos-monkey-playing-ping-pong