Using Channels to Communicate Between Coroutines

Kevin Jones

@kevinrjones www.rocksolidknowledge.com

Channels

Why channels?

What are channels?

Creating channels

Blocking channels

Buffered channels

Synchronization

Iteration and closing channels

Why Channels?

Single coroutines can communicate by returning a deferred

Channels offer a more generalized communication mechanism

What Are Channels?

Use channels to communicate between coroutines

- Send to and receive from a channel
- More than one item of data
- Channels block
- Can create buffered channels
- Need to know when channel has finished

```
val channel = Channel<Int>()


channel.send(1)
var value = channel.receive()
```

Creating Channels

Channels are typed

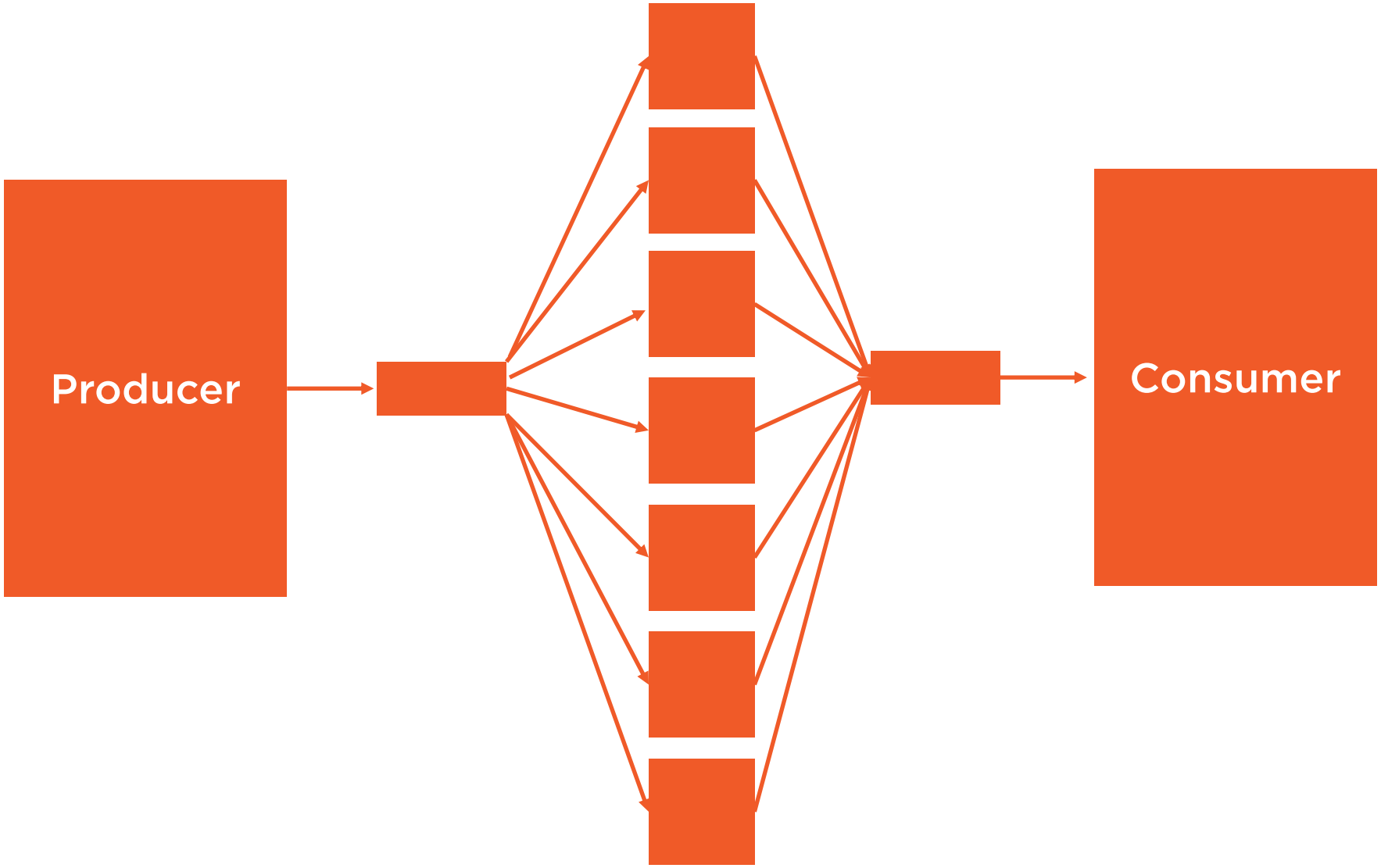
Send and Receive both suspend

Channels are a rendezvous point

Demo

Using channels

Load Balancer



Demo

Bringing it together – a simple load balancer

Summary

Channels provide a mechanism for coroutines to communicate

- They need managing
- Blocked channels can be an issue
- Close channels when finished

What's Next