

# Managing an Akka Cluster on Kubernetes

7-8

**Ridesharing service**  
at MOIA

3-6

**ConductR**  
at Lightbend

1-3

**Test automation service**  
at Testmunk





MOIA ❤️ Scala

# Disclaimer

[A career in distributed systems] is both exhilarating and frustrating. When things work, it's like a symphony. When they don't, it's like an eleventh-birthday party where half of the kids are on speed.

– Jeff Darcy, HekaFS (formerly CloudFS) project lead

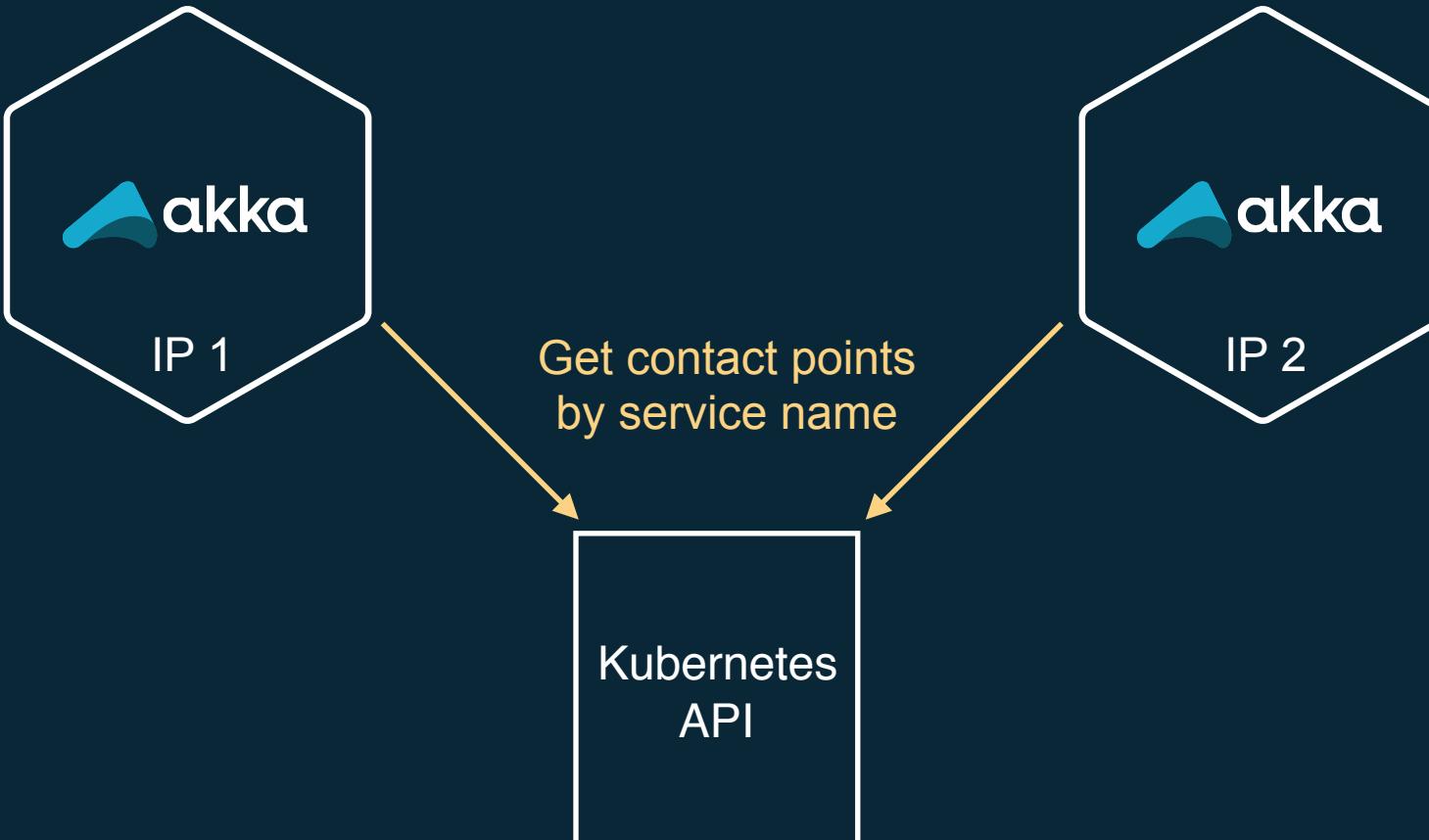
# Bootstrap



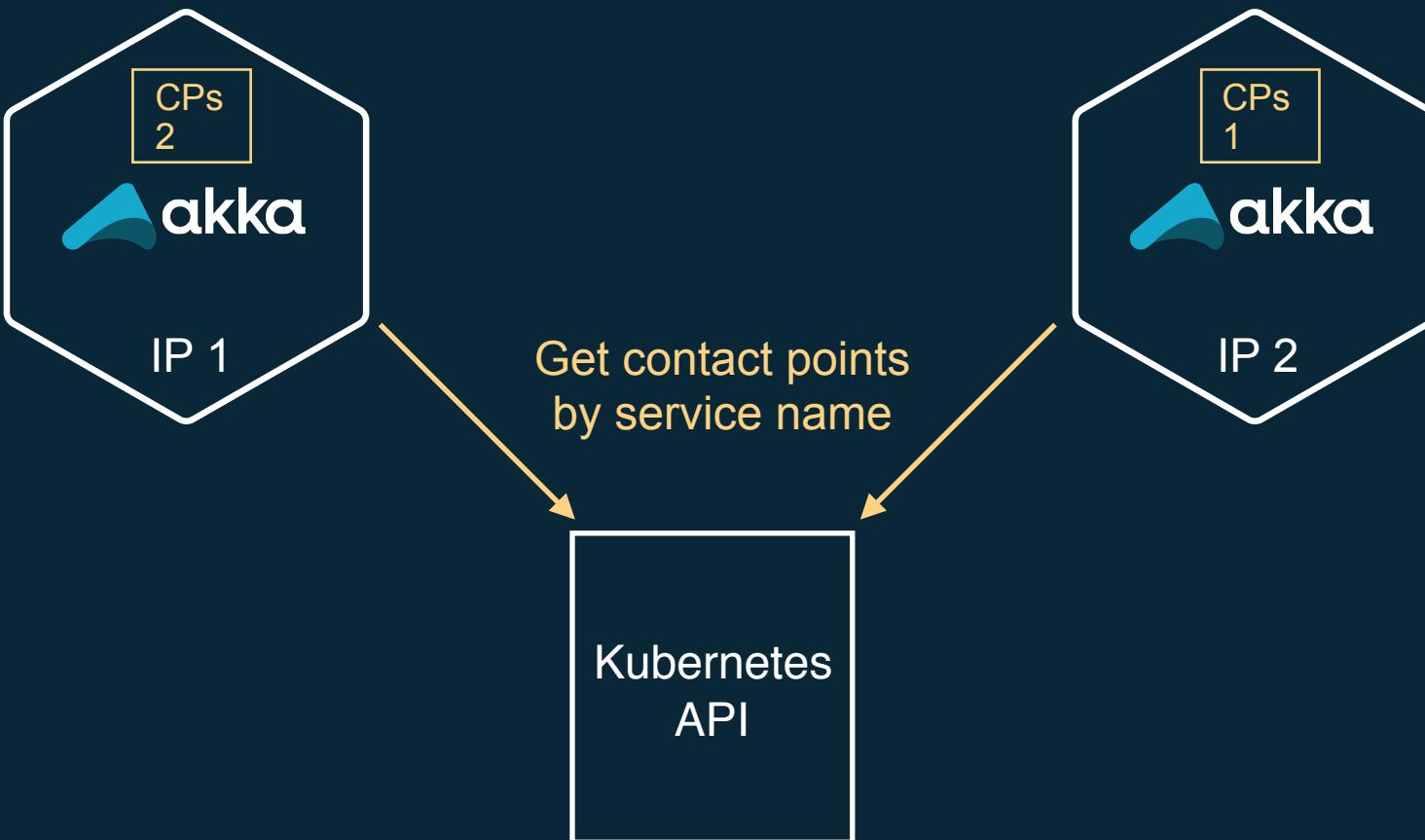
# Contact point discovery



# Contact point discovery



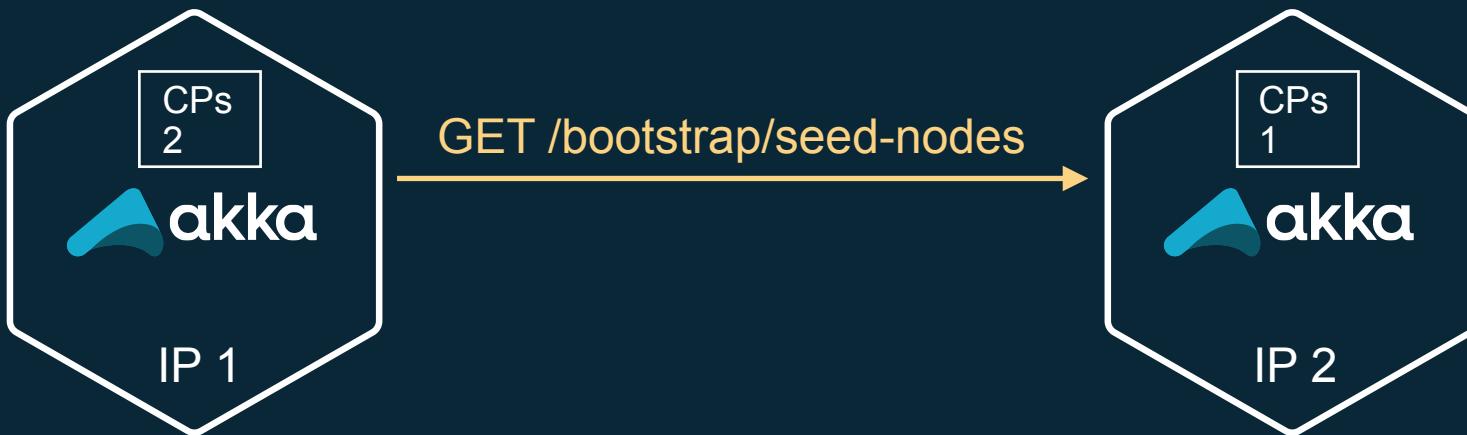
# Contact point discovery



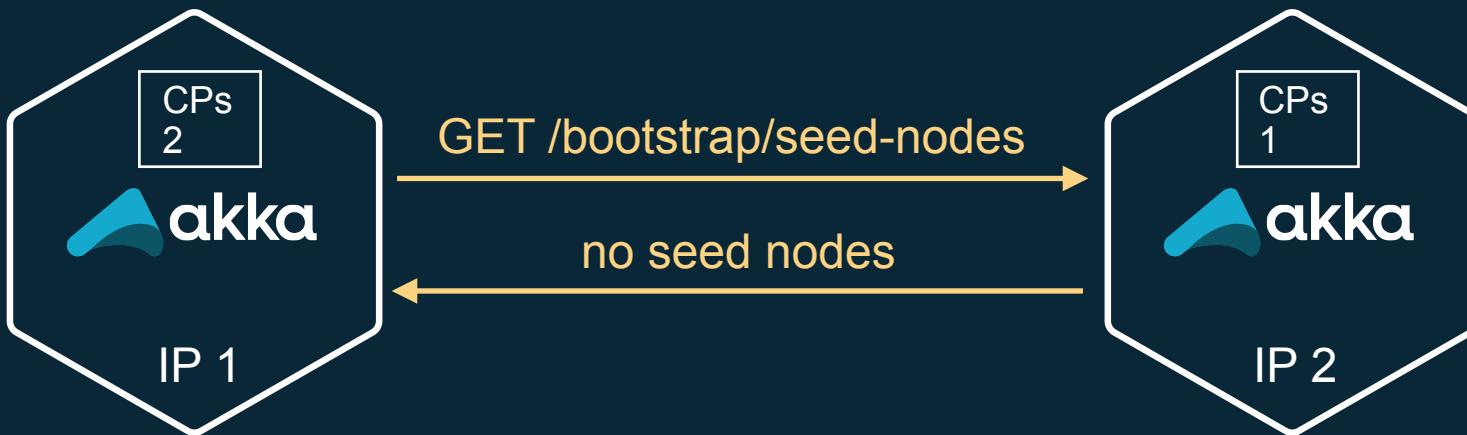
# Contact probing



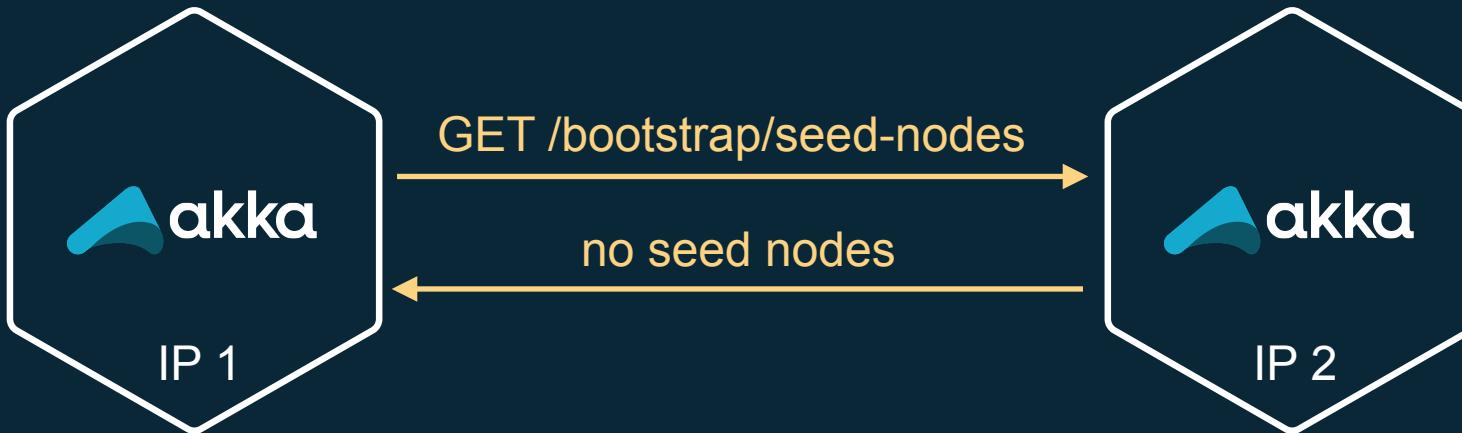
# Contact probing



# Contact probing



# Initial joining



# Initial joining



# Initial joining



stable time margin passed



# Initial joining



Self join

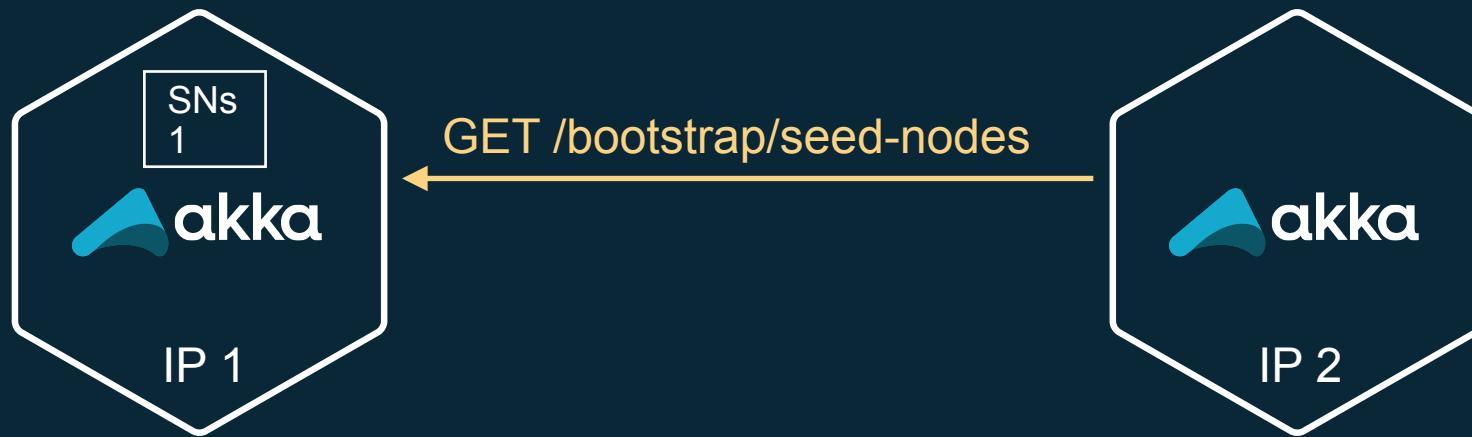
stable time margin passed



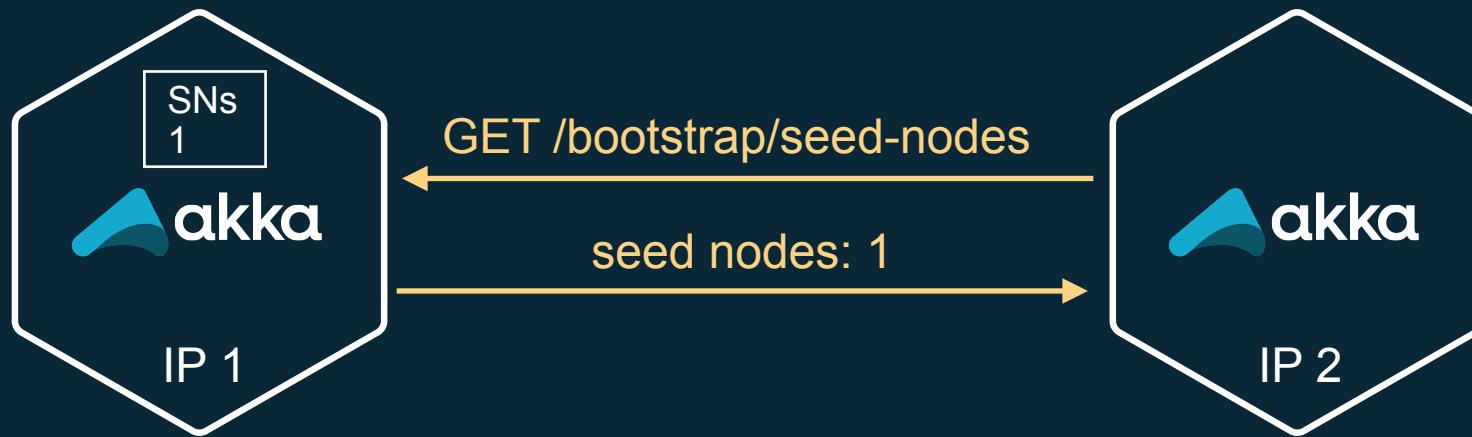
# Joining existing cluster



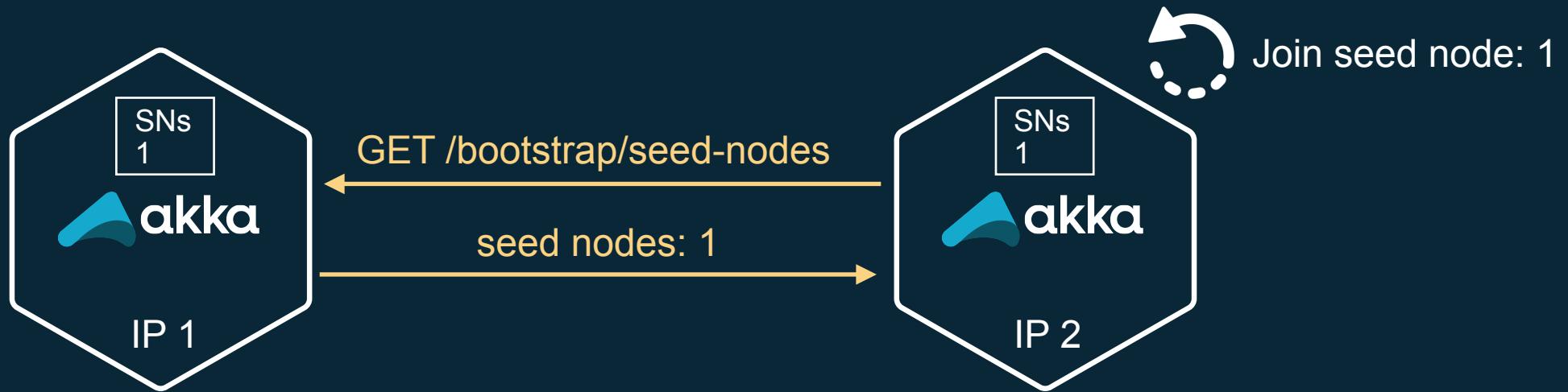
# Joining existing cluster



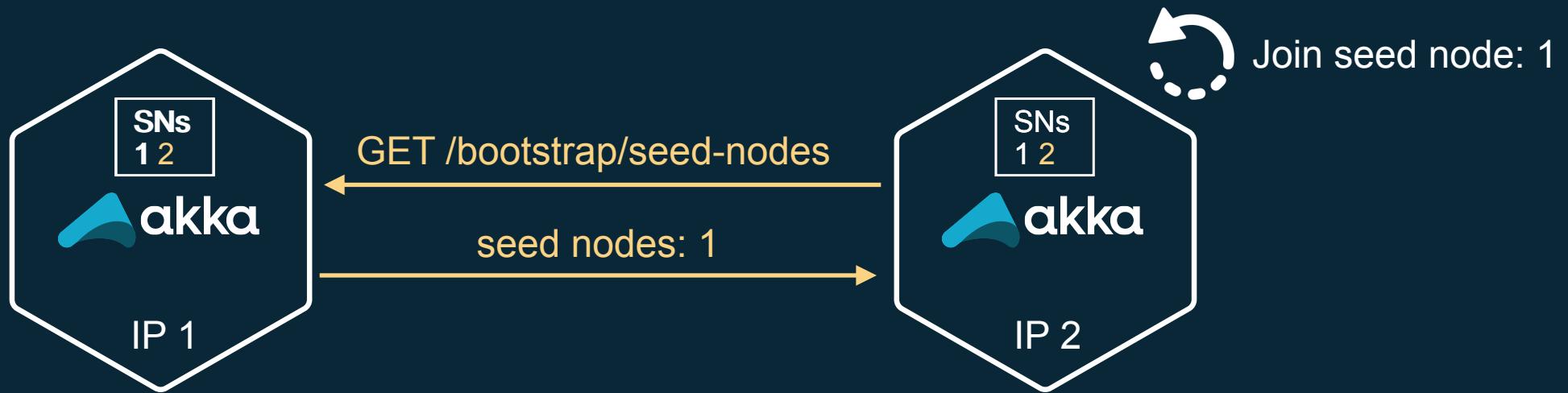
# Joining existing cluster



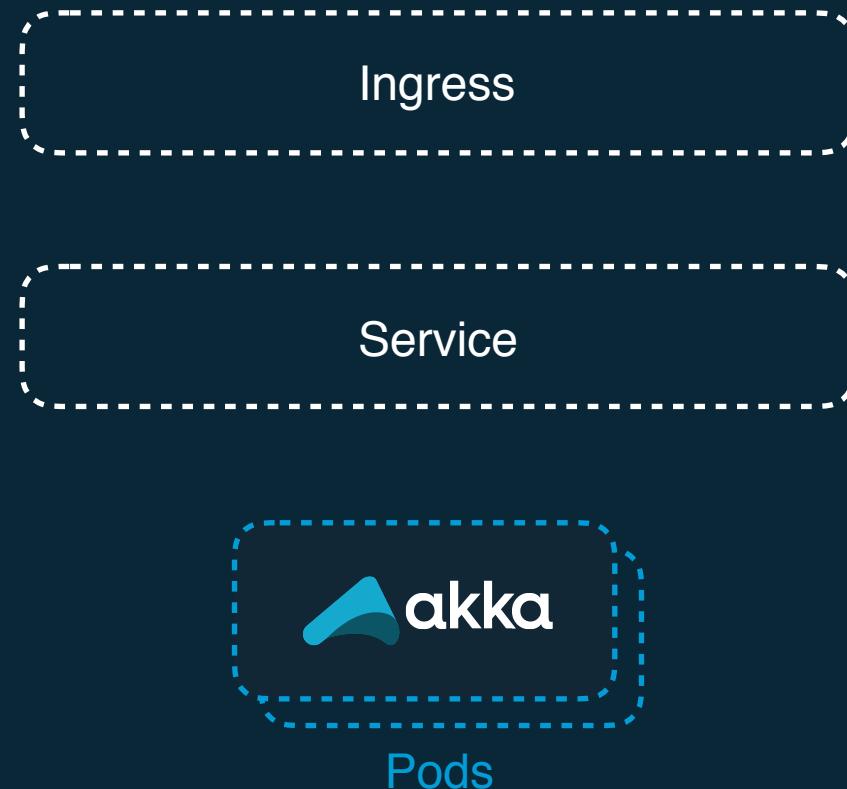
# Joining existing cluster



# Joining existing cluster



## Kubernetes Cluster



# Demo



# Shutdown

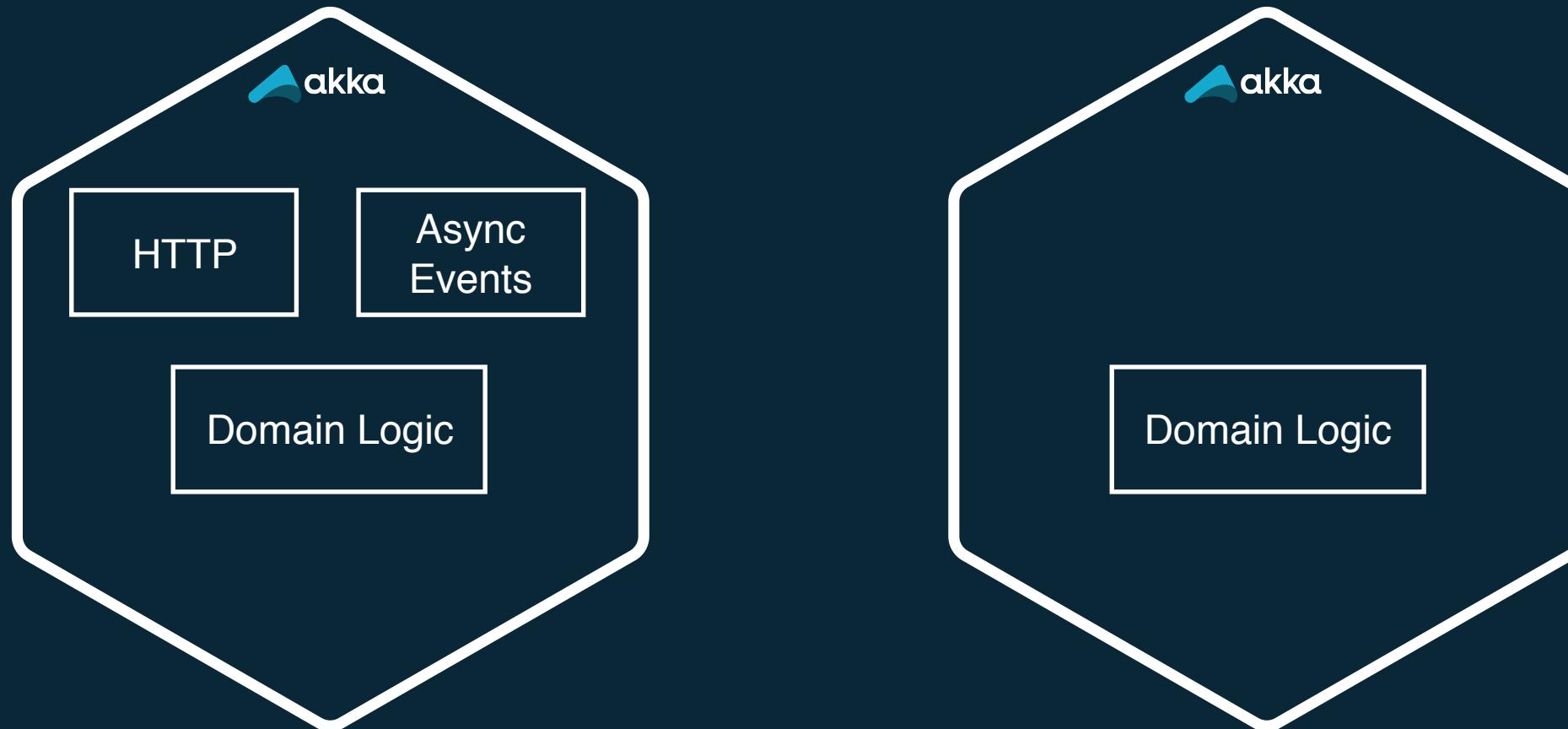




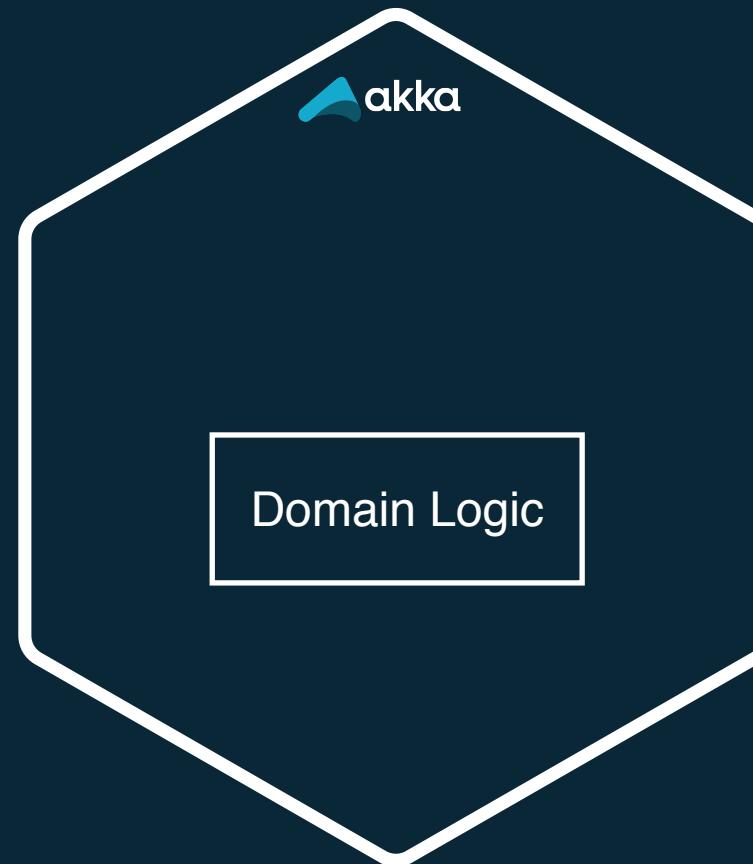
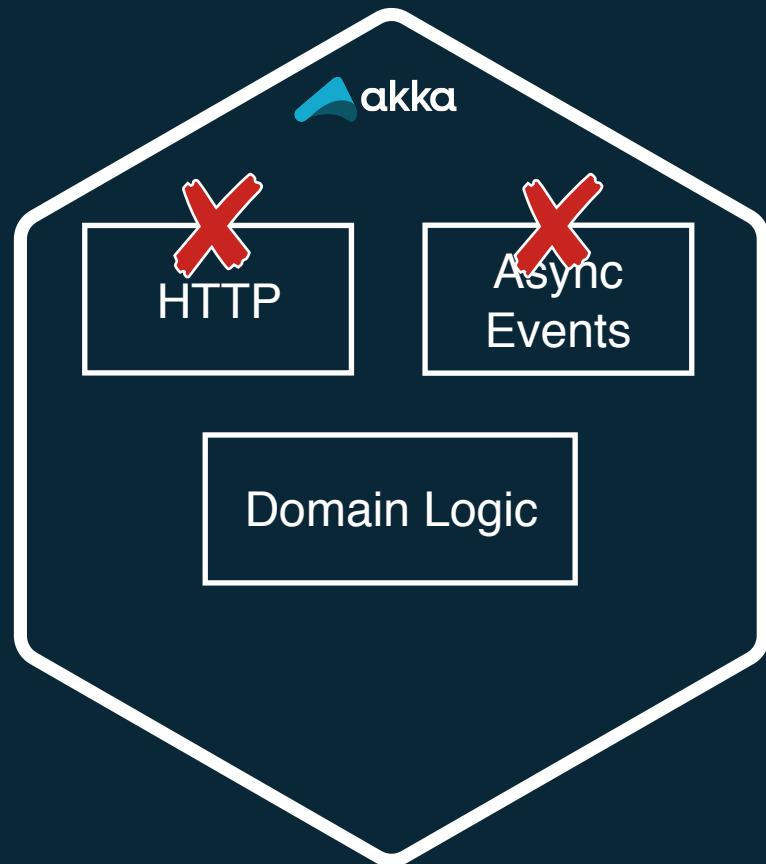
# Coordinated Shutdown



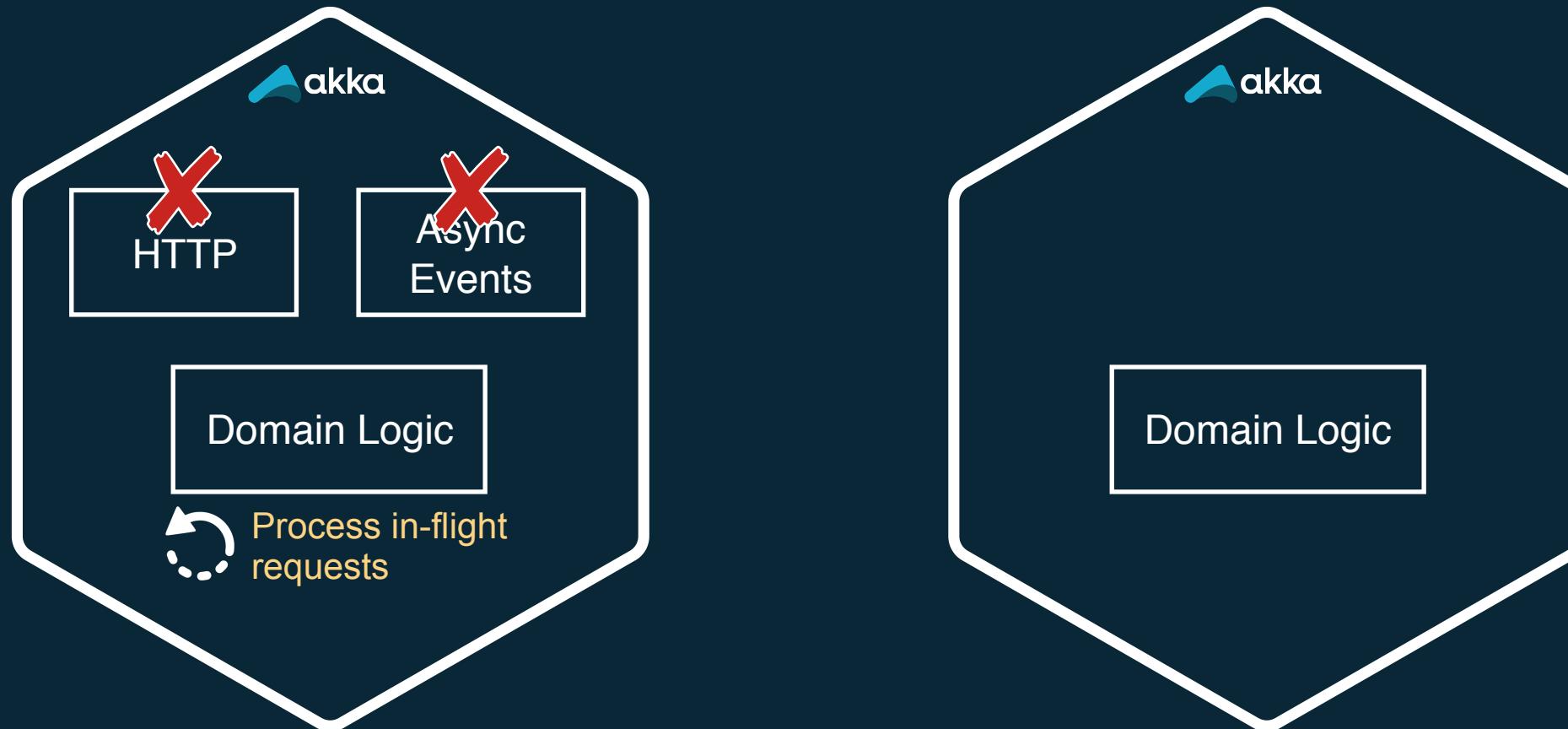
# This is easy, right?



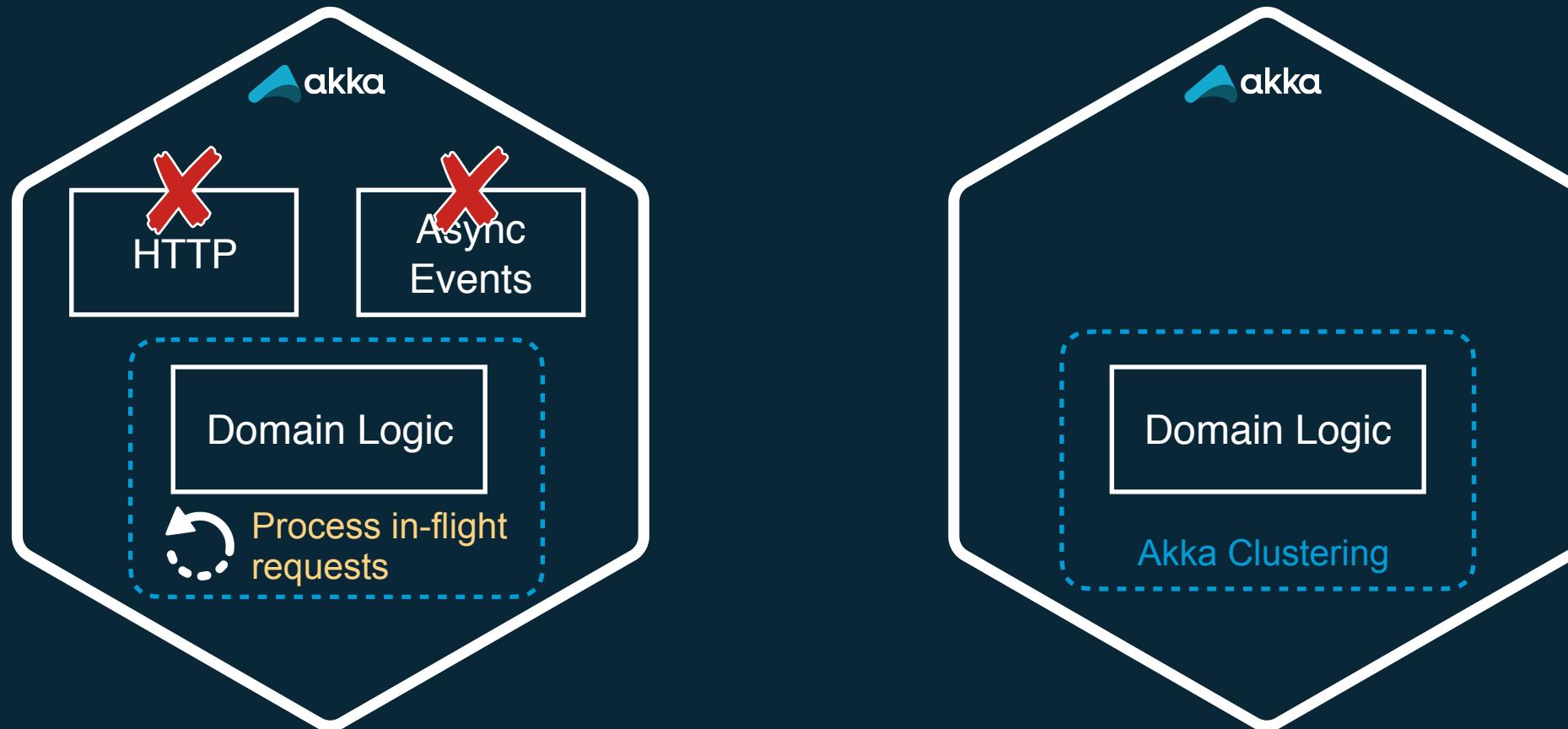
# This is easy, right?



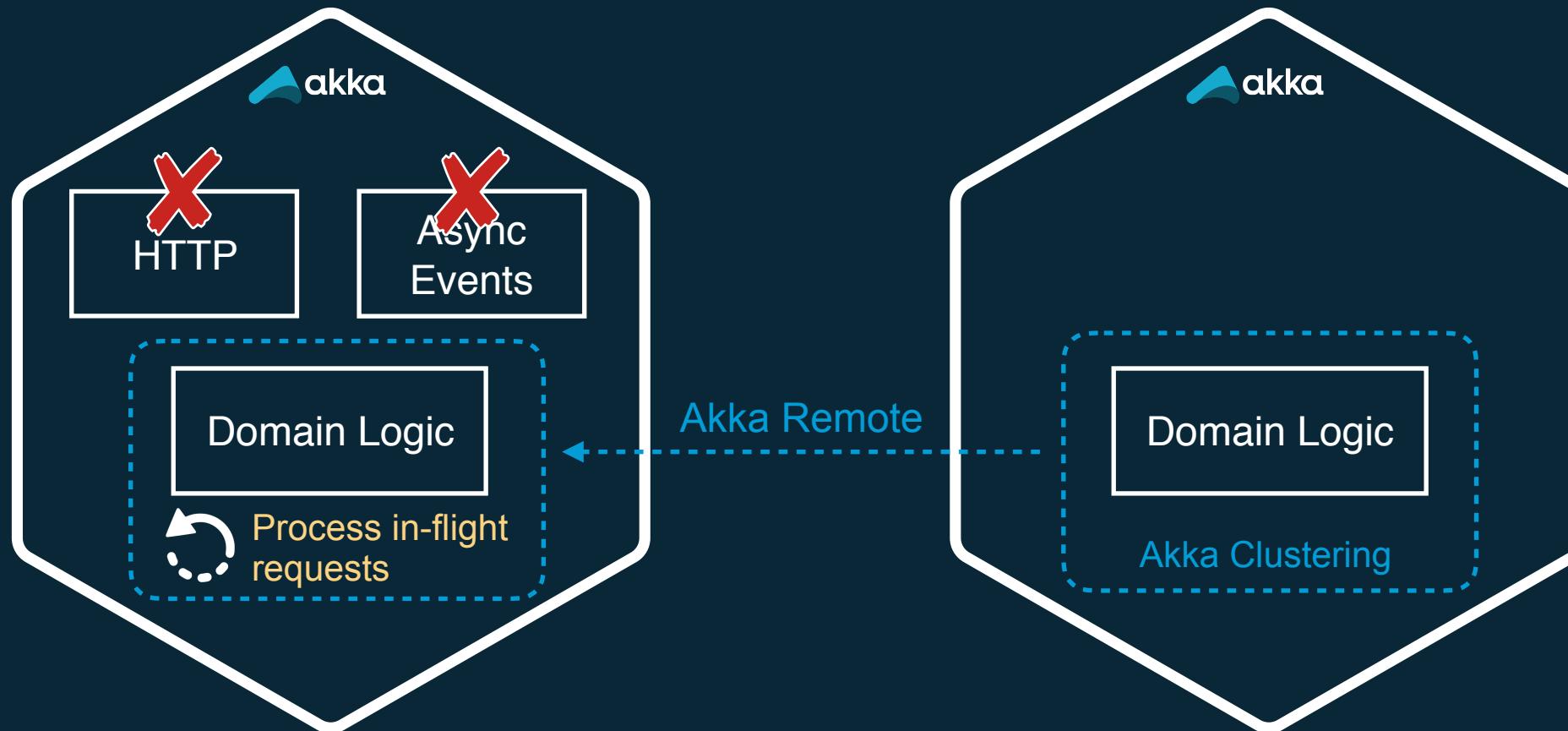
# This is easy, right?



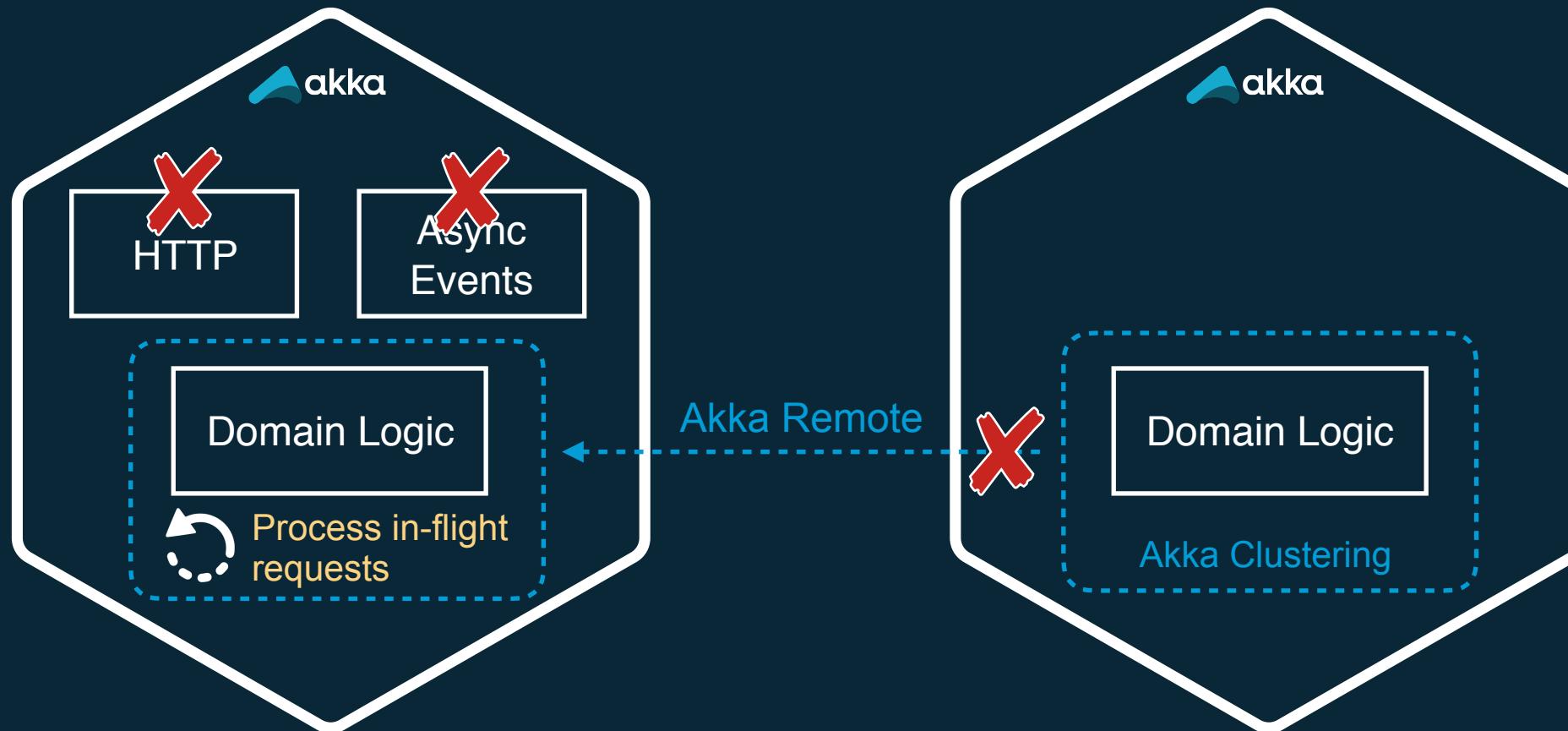
# This is easy, right?



# This is easy, right?



# This is easy, right?



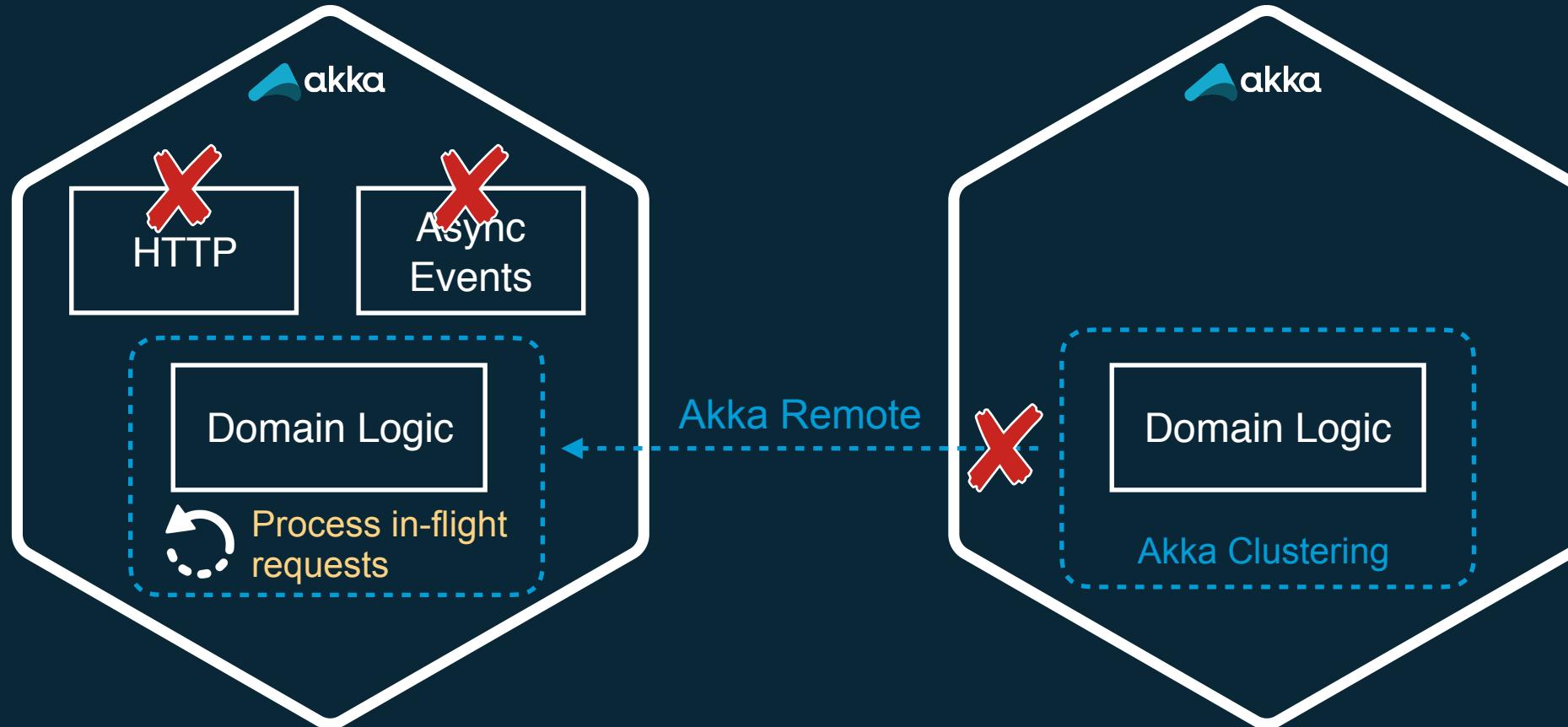


Heiko Seeberger  
@hseeberger

Akka Streams to the Extreme

11:15-12:00 - Wednesday

# This is easy, right?



# Akka Coordinated Shutdown phases

01 BeforeServiceUnbind

custom phase

# Akka Coordinated Shutdown phases

- |    |                     |                                  |
|----|---------------------|----------------------------------|
| 01 | BeforeServiceUnbind | custom phase                     |
| 02 | ServiceUnbind       | stop accepting incoming requests |

# Akka Coordinated Shutdown phases

- |    |                     |                                  |
|----|---------------------|----------------------------------|
| 01 | BeforeServiceUnbind | custom phase                     |
| 02 | ServiceUnbind       | stop accepting incoming requests |
| 03 | ServiceRequestsDone | process in-flight requests       |

# Akka Coordinated Shutdown phases

01	BeforeServiceUnbind	custom phase
02	ServiceUnbind	stop accepting incoming requests
03	ServiceRequestsDone	process in-flight requests
06	ClusterShardingShutdownRegion	cluster sharding shutdown

# Akka Coordinated Shutdown phases

01	BeforeServiceUnbind	custom phase
02	ServiceUnbind	stop accepting incoming requests
03	ServiceRequestsDone	process in-flight requests
06	ClusterShardingShutdownRegion	cluster sharding shutdown
08	ClusterExiting	cluster singleton shutdown

# Akka Coordinated Shutdown phases

01	BeforeServiceUnbind	custom phase
02	ServiceUnbind	stop accepting incoming requests
03	ServiceRequestsDone	process in-flight requests
06	ClusterShardingShutdownRegion	cluster sharding shutdown
08	ClusterExiting	cluster singleton shutdown
12	ActorSystemTerminate	terminate actor system & JVM

# Register stop logic

```
import akka.actor.CoordinatedShutdown
import akka.actor.CoordinatedShutdown.PhaseBeforeClusterShutdown

CoordinatedShutdown(system)
  .addTask(PhaseBeforeClusterShutdown, "my.module") { () =>
    // Add graceful stop logic here
}
```

# Trigger coordinated shutdown

```
import akka.actor.CoordinatedShutdown.Reason  
  
private final case object MyShutdownReason extends Reason  
  
CoordinatedShutdown(system).run(MyShutdownReason)
```

# Akka HTTP shutdown

```
Http()  
  .bindAndHandle(route, address, port)  
  .onComplete {  
    case Failure(_) =>  
      logger.error(s"Shutting down because cannot bind to $address:$port")  
      shutdown.run(BindFailure)  
  
    case Success(binding) =>  
      logger.info(s"Listening for HTTP connections on ${binding.localAddress}")  
      shutdown.addTask(PhaseServiceRequestsDone, "api.terminate") { () =>  
        binding.terminate(terminationDeadline).map(_ => Done)  
      }  
  }
```

# Putting things together

Kubernetes



# Putting things together

1. Signal shutdown

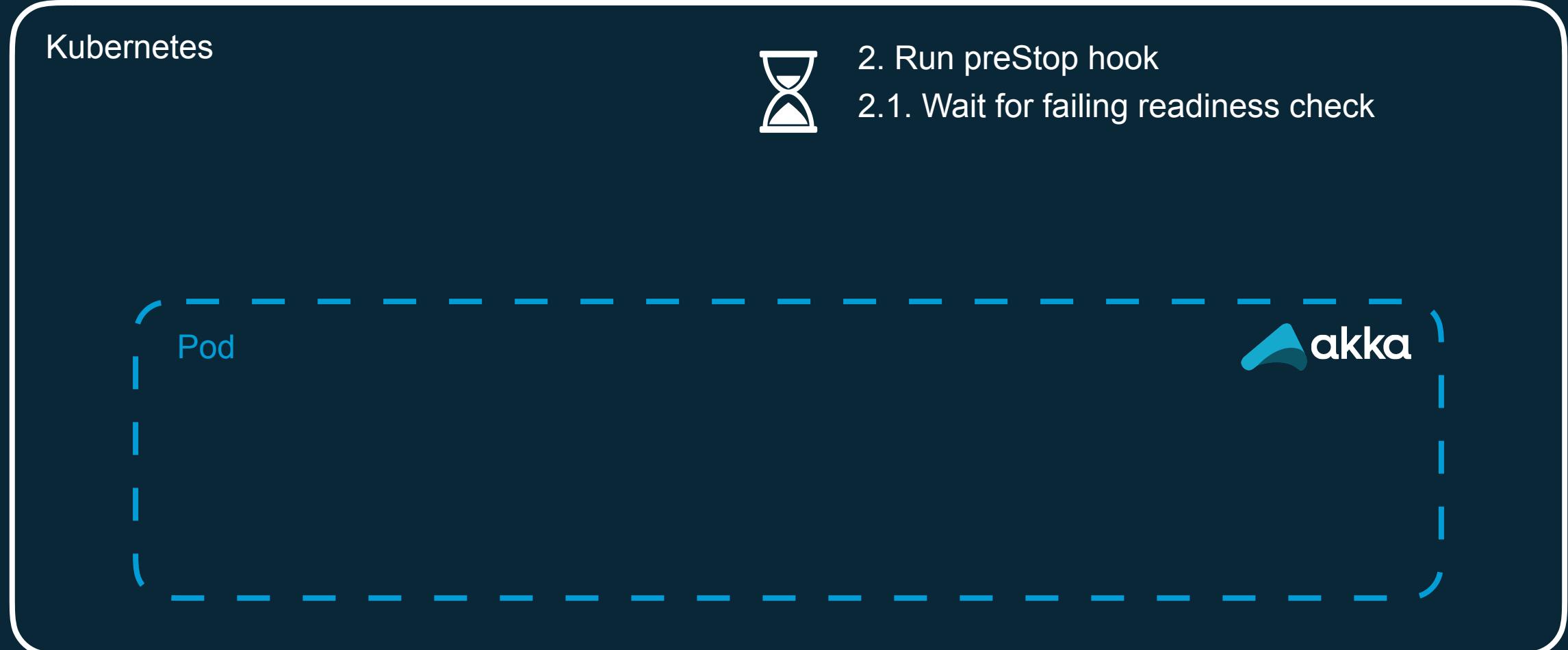
Kubernetes



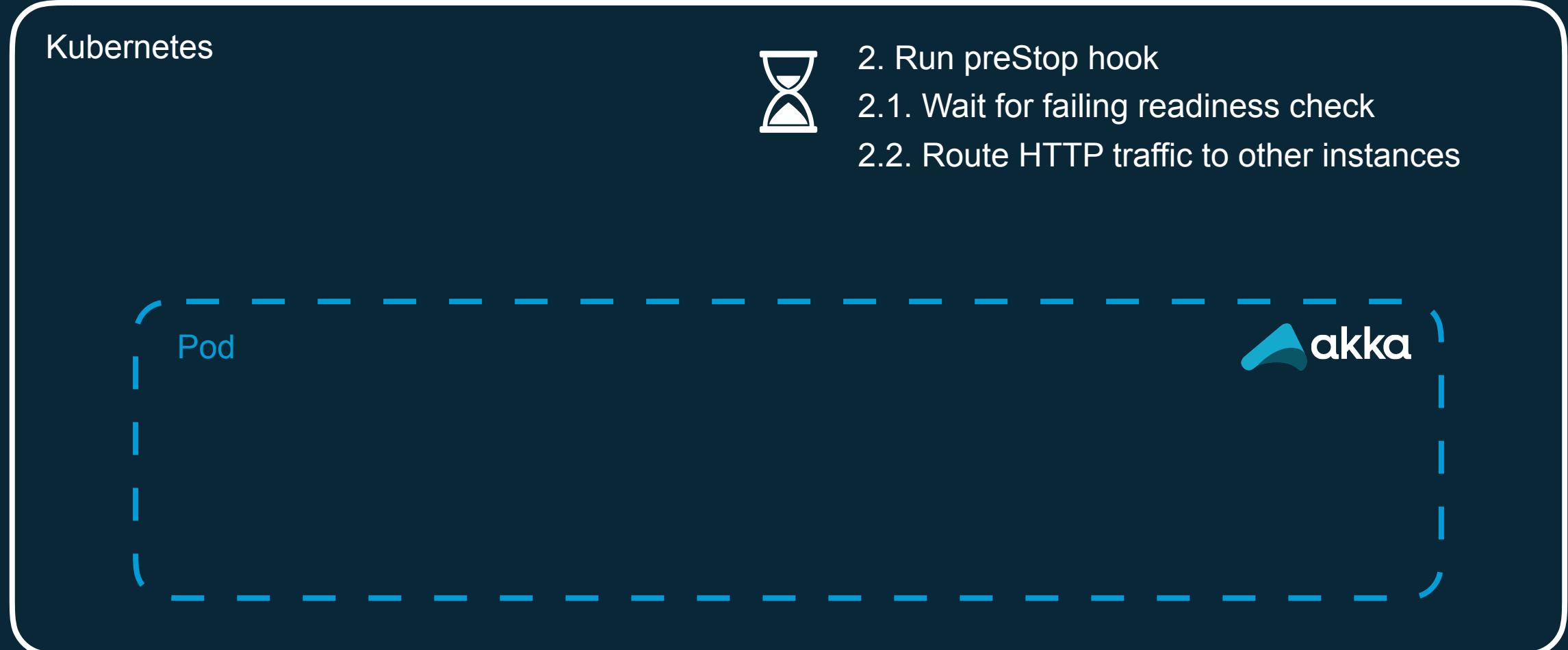
# Putting things together



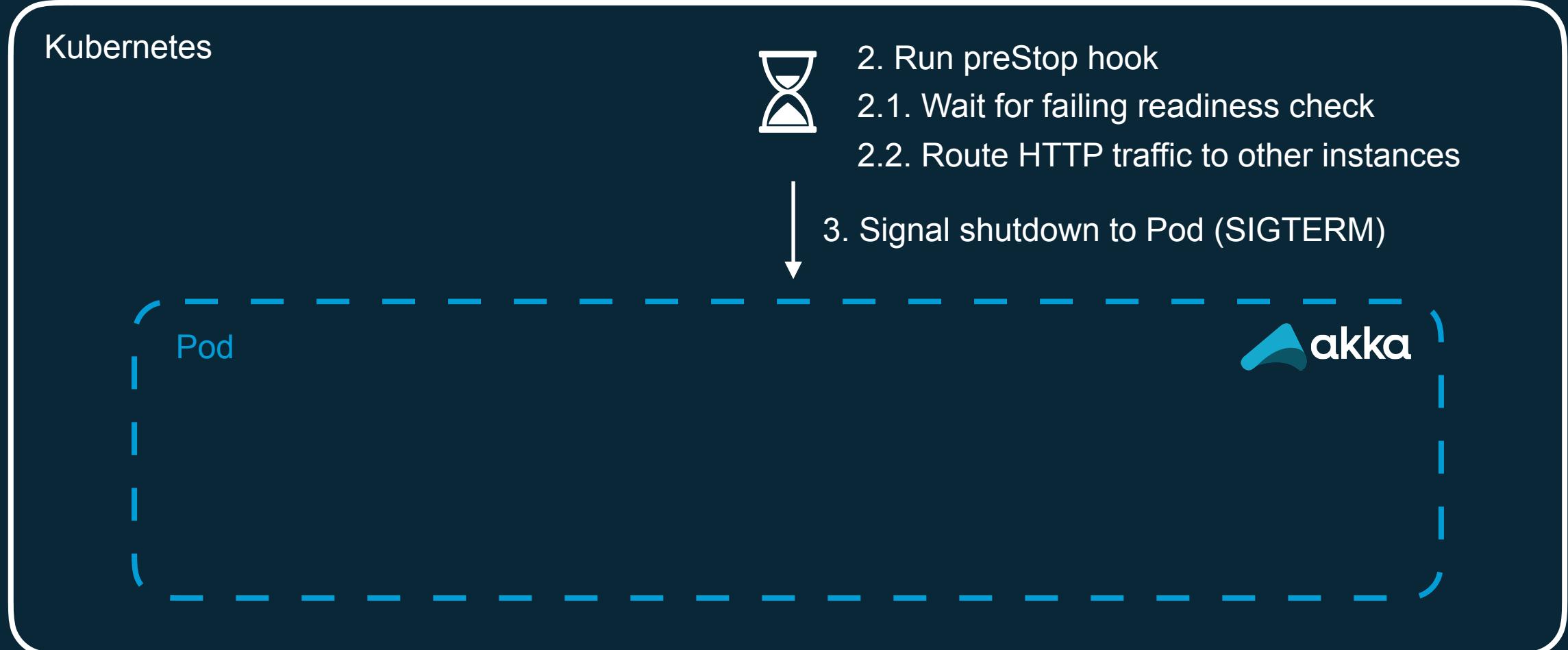
# Putting things together



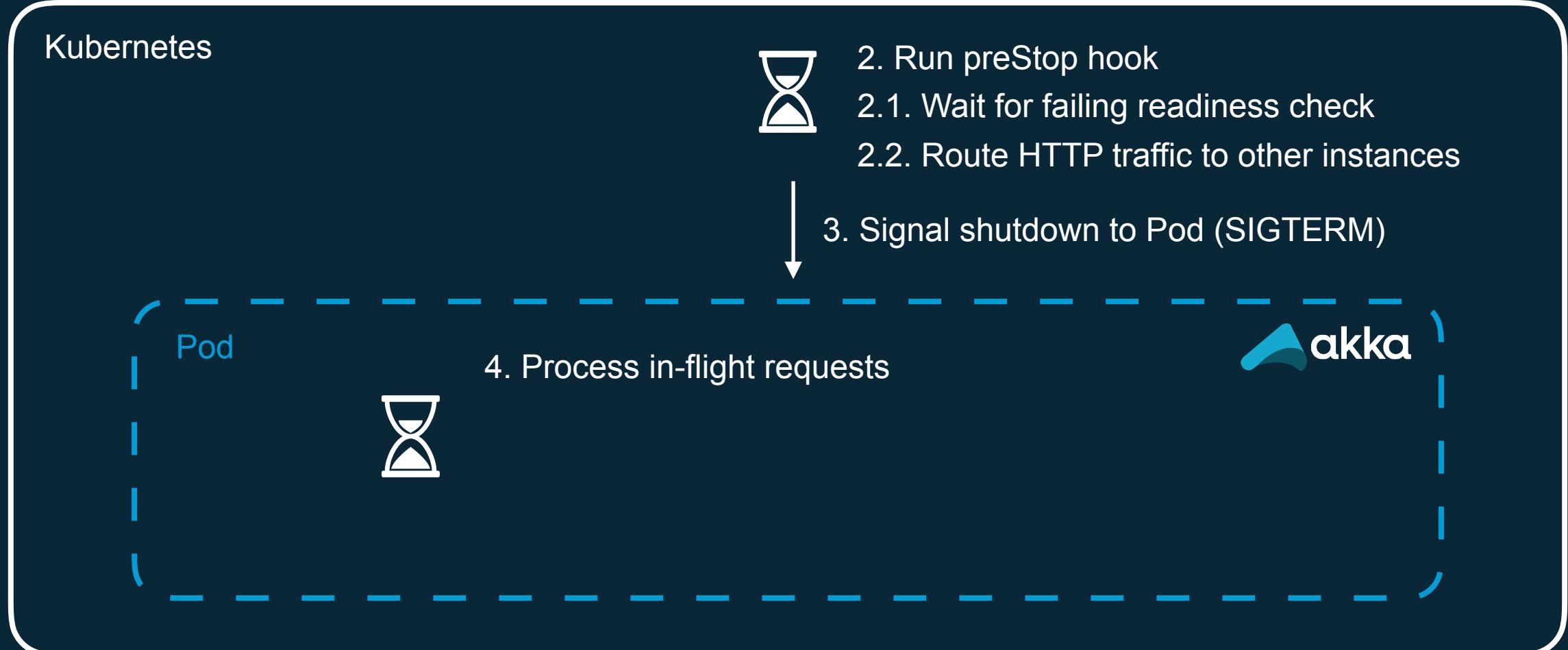
# Putting things together



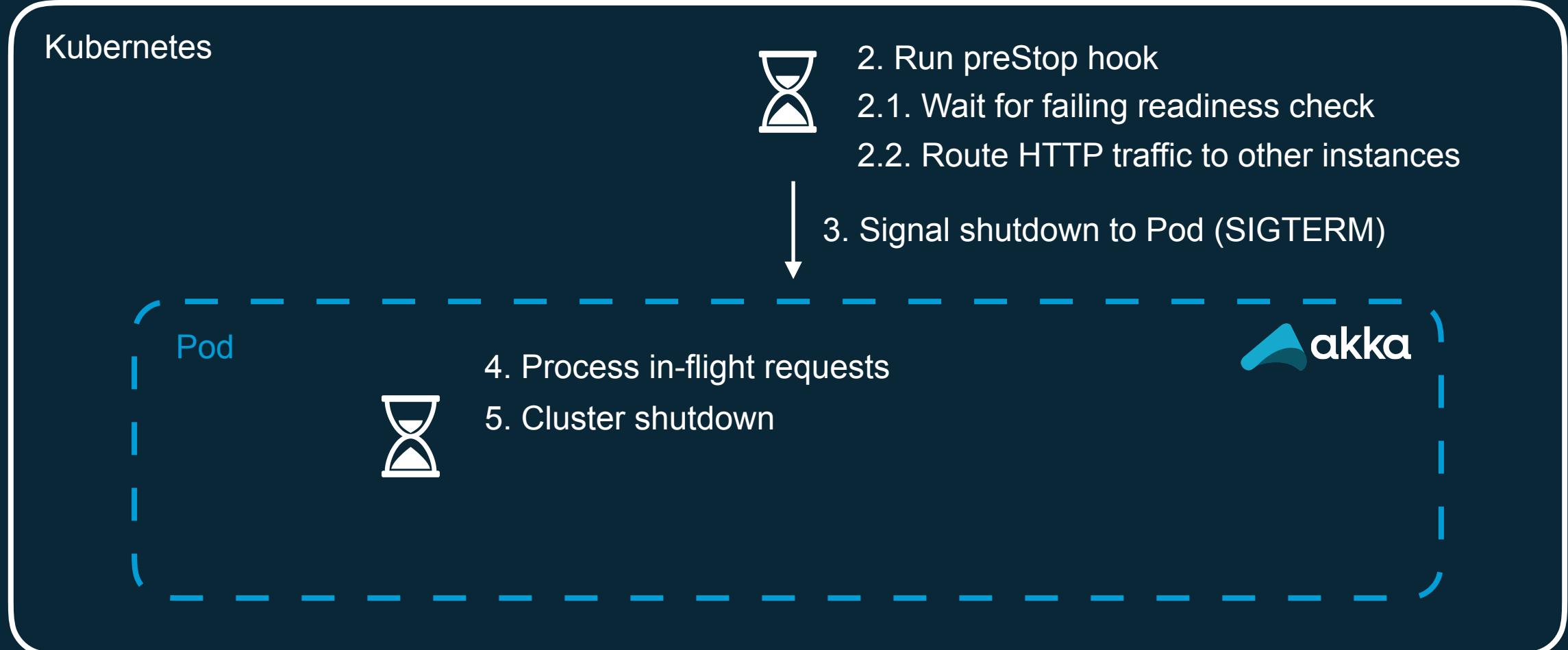
# Putting things together



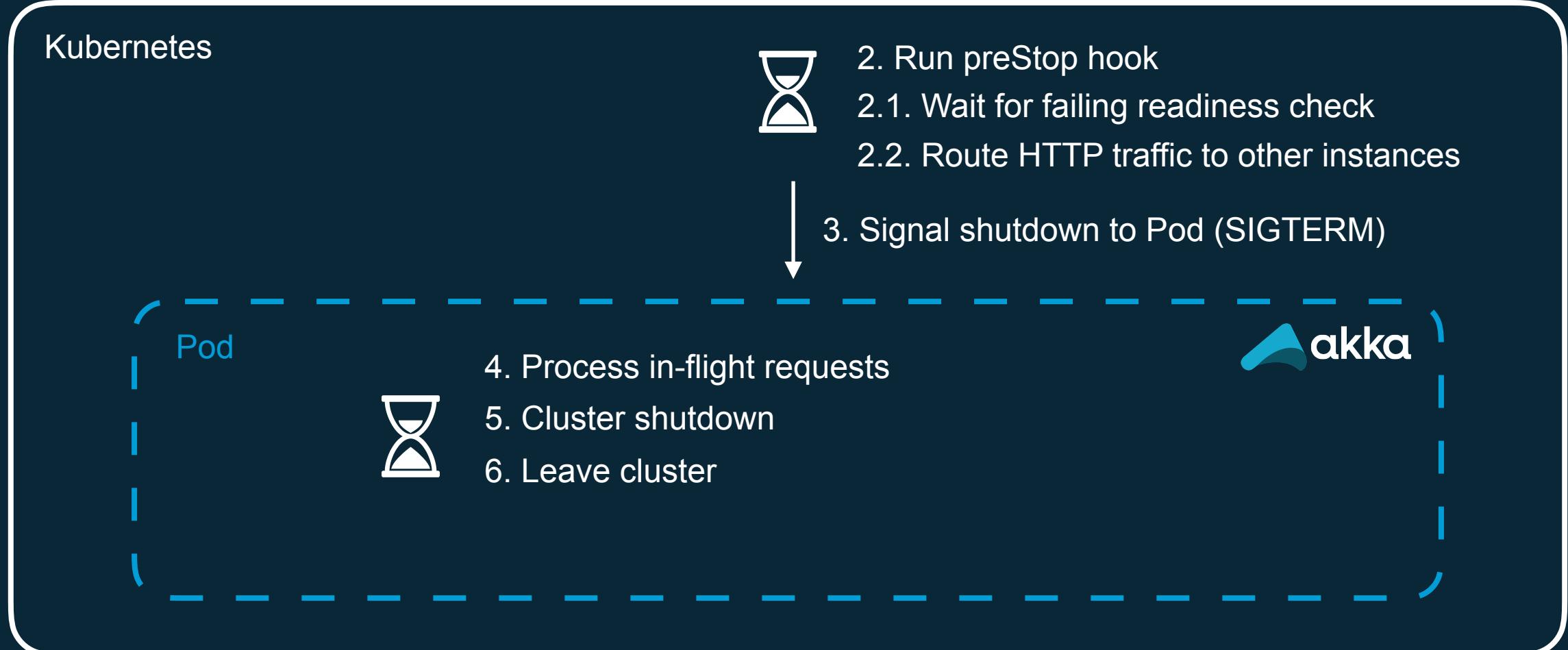
# Putting things together



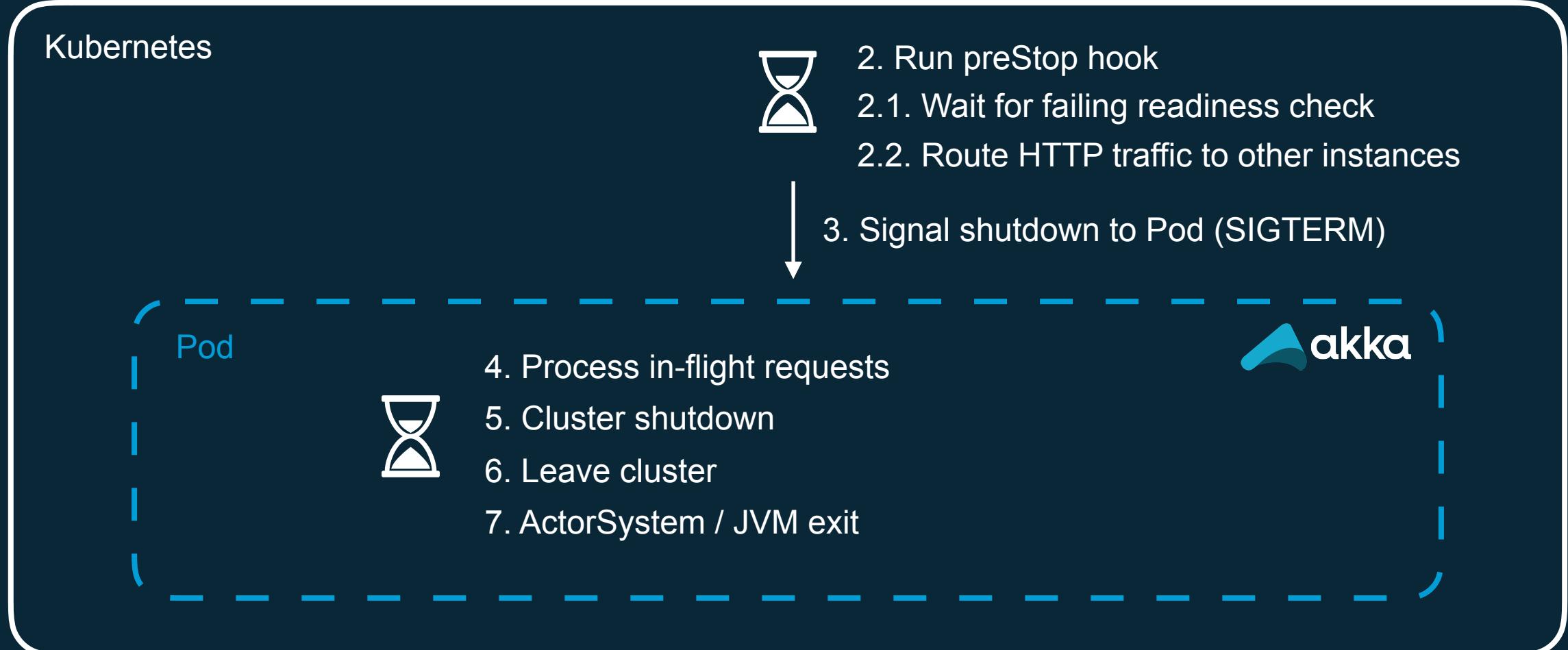
# Putting things together



# Putting things together



# Putting things together



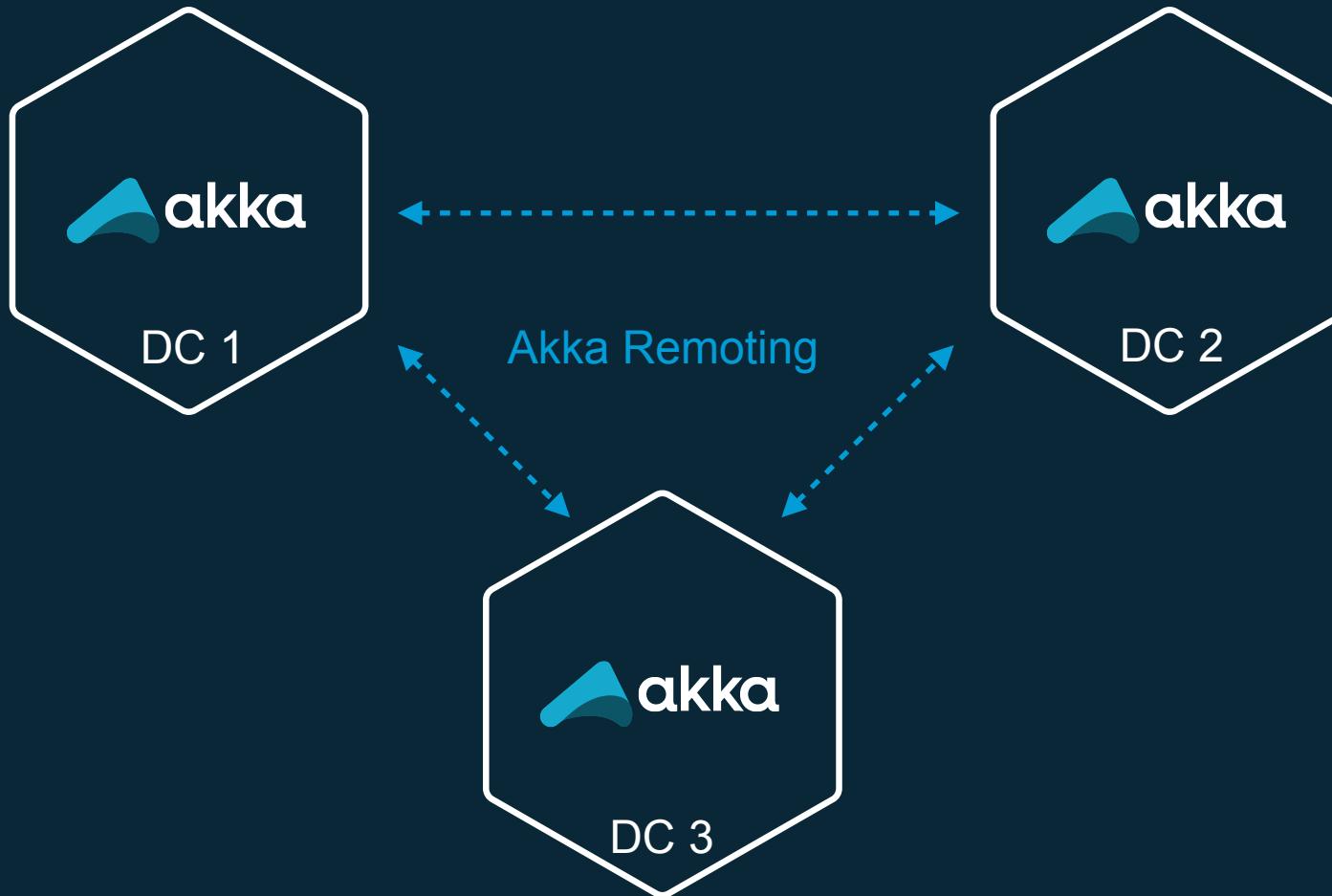


Demo

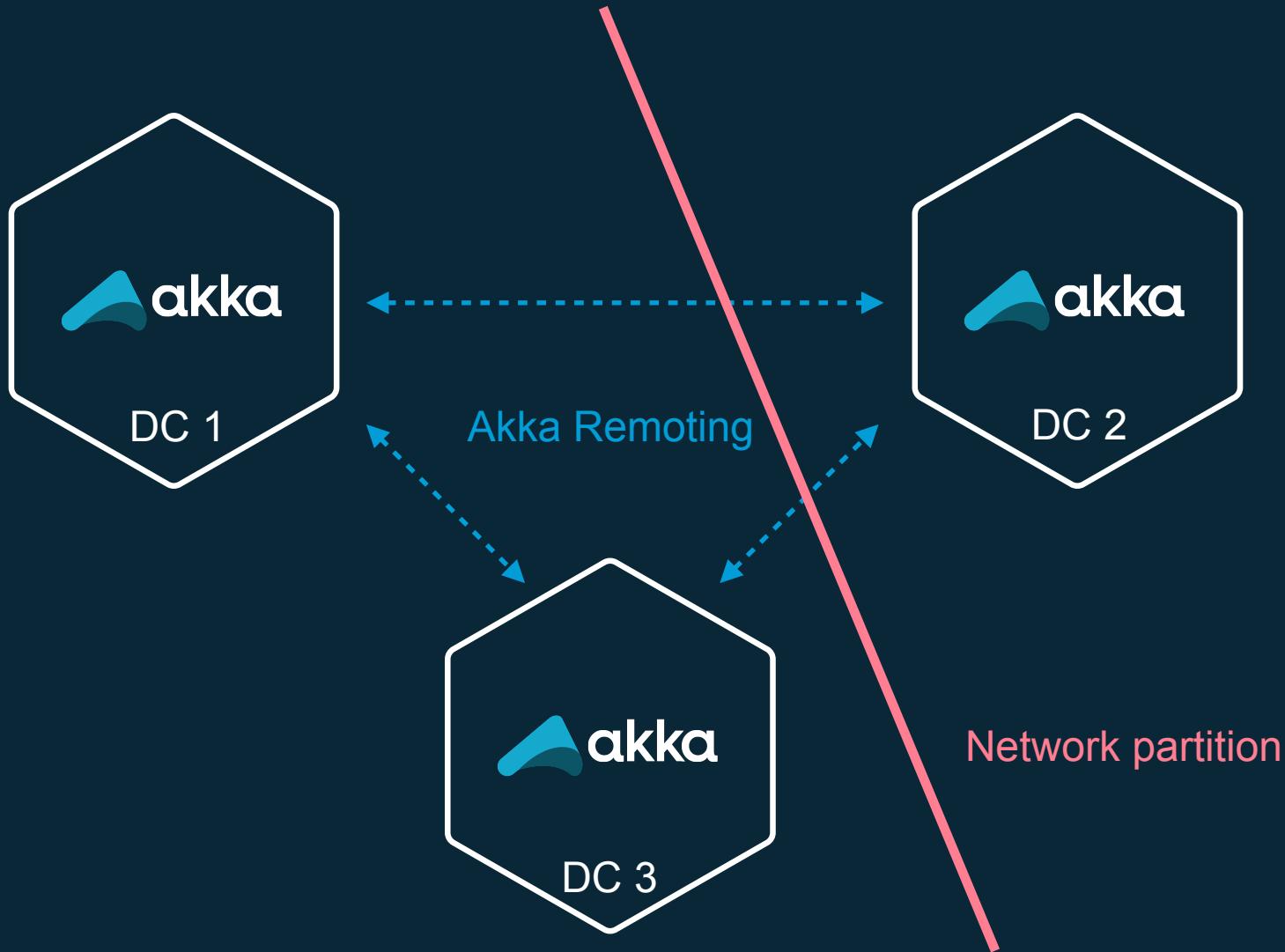
Split brain  
resolution



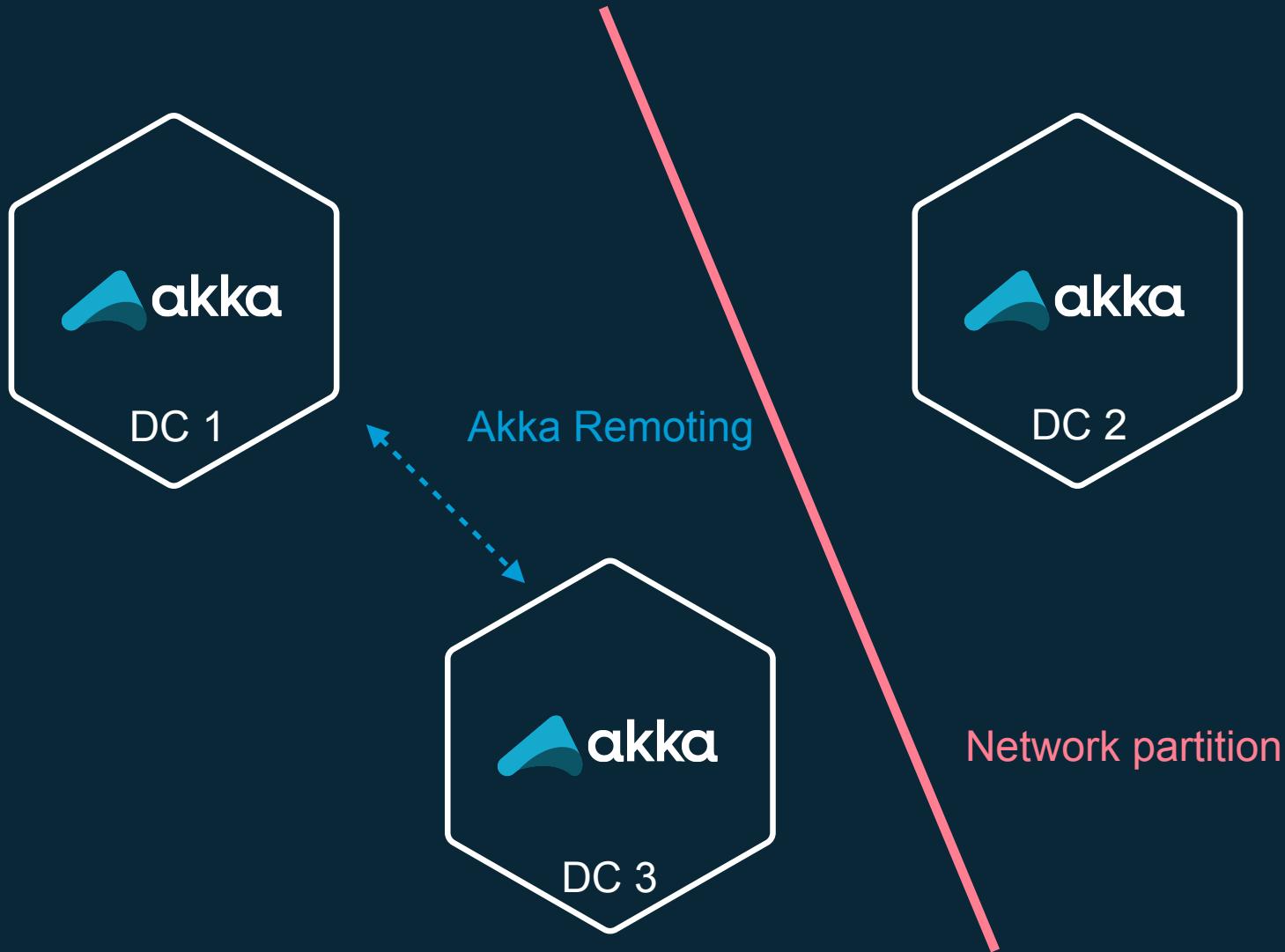
# Network partition



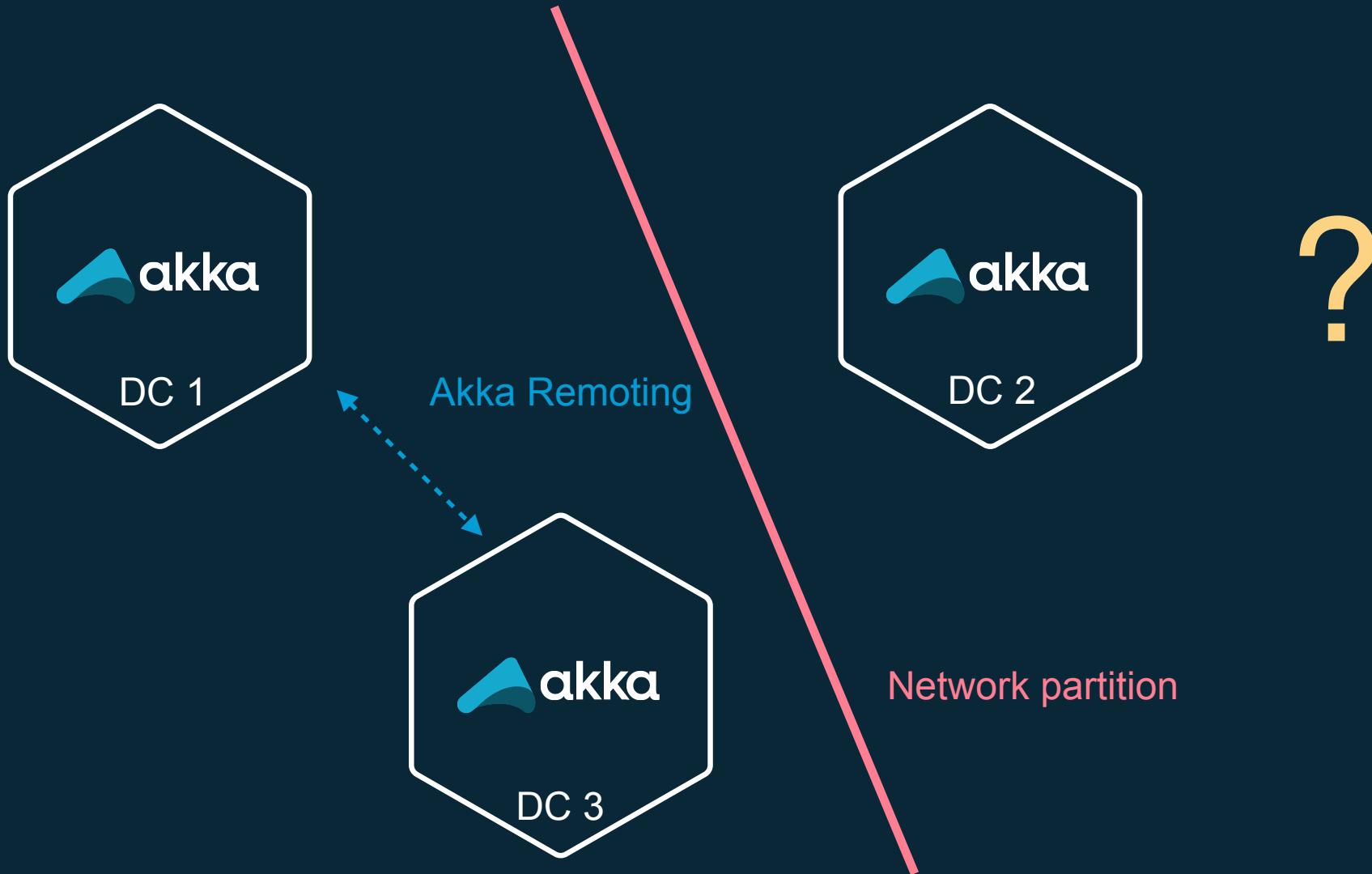
# Network partition



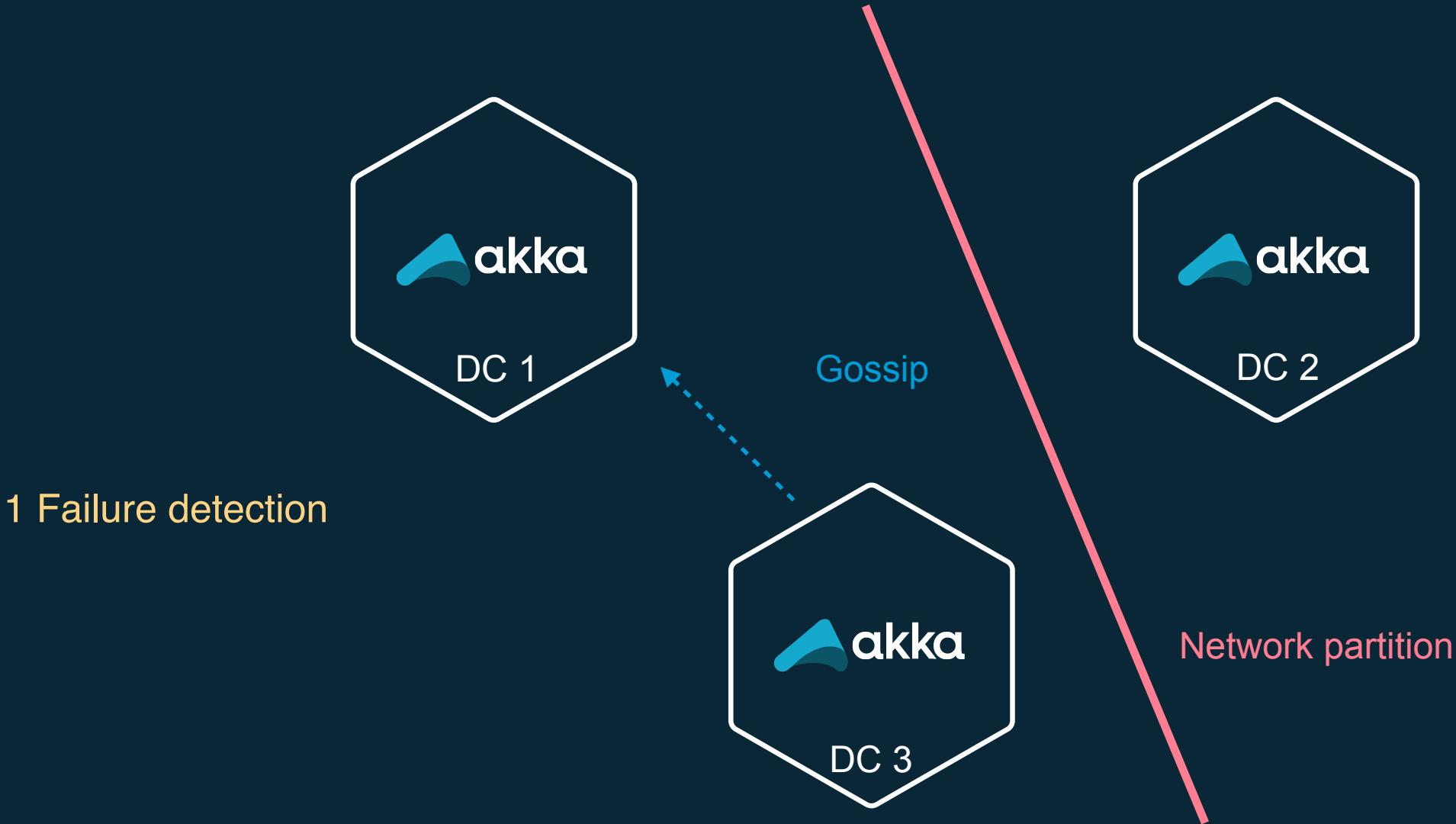
# Network partition



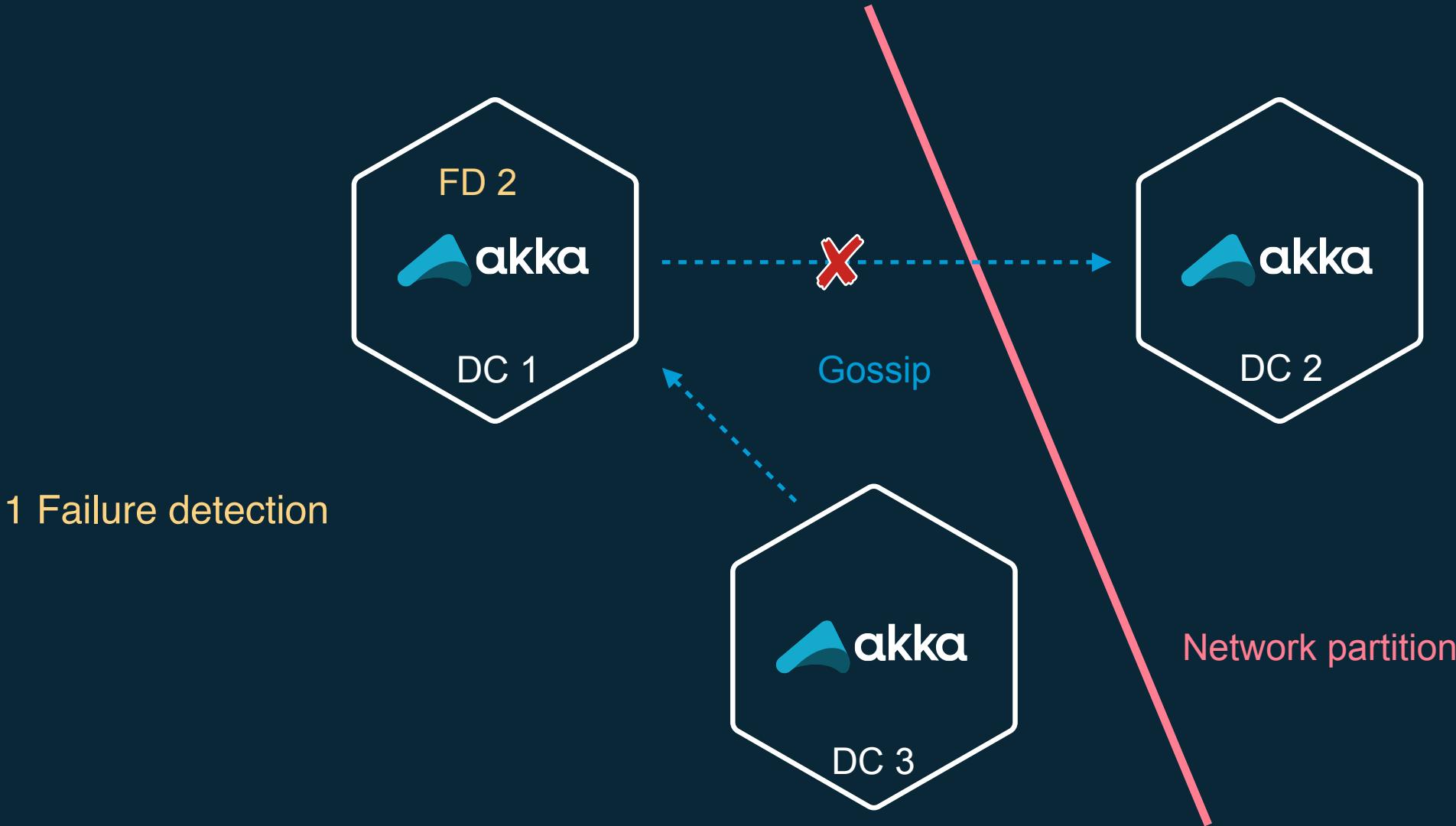
# Network partition



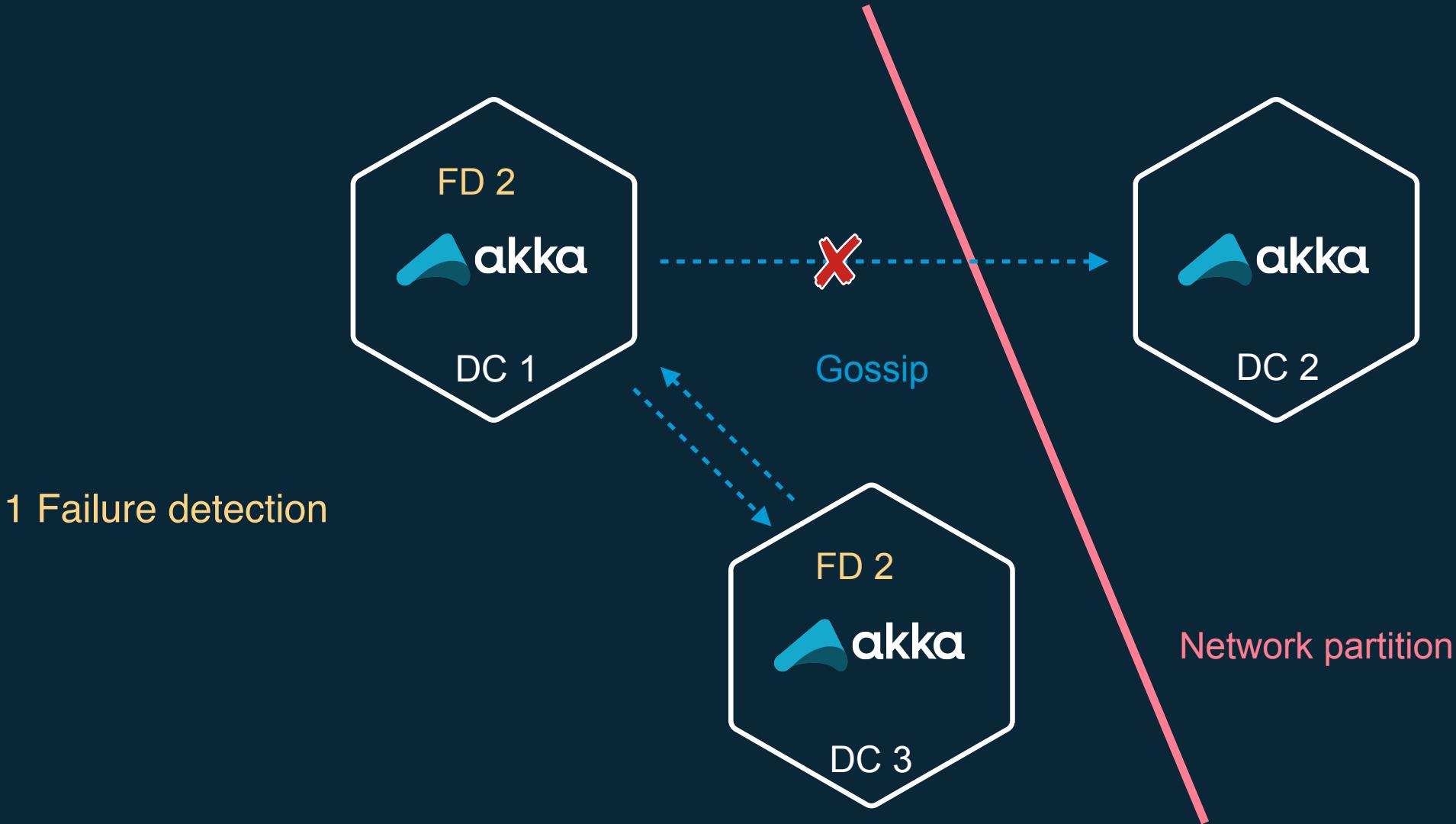
# Handling network partition



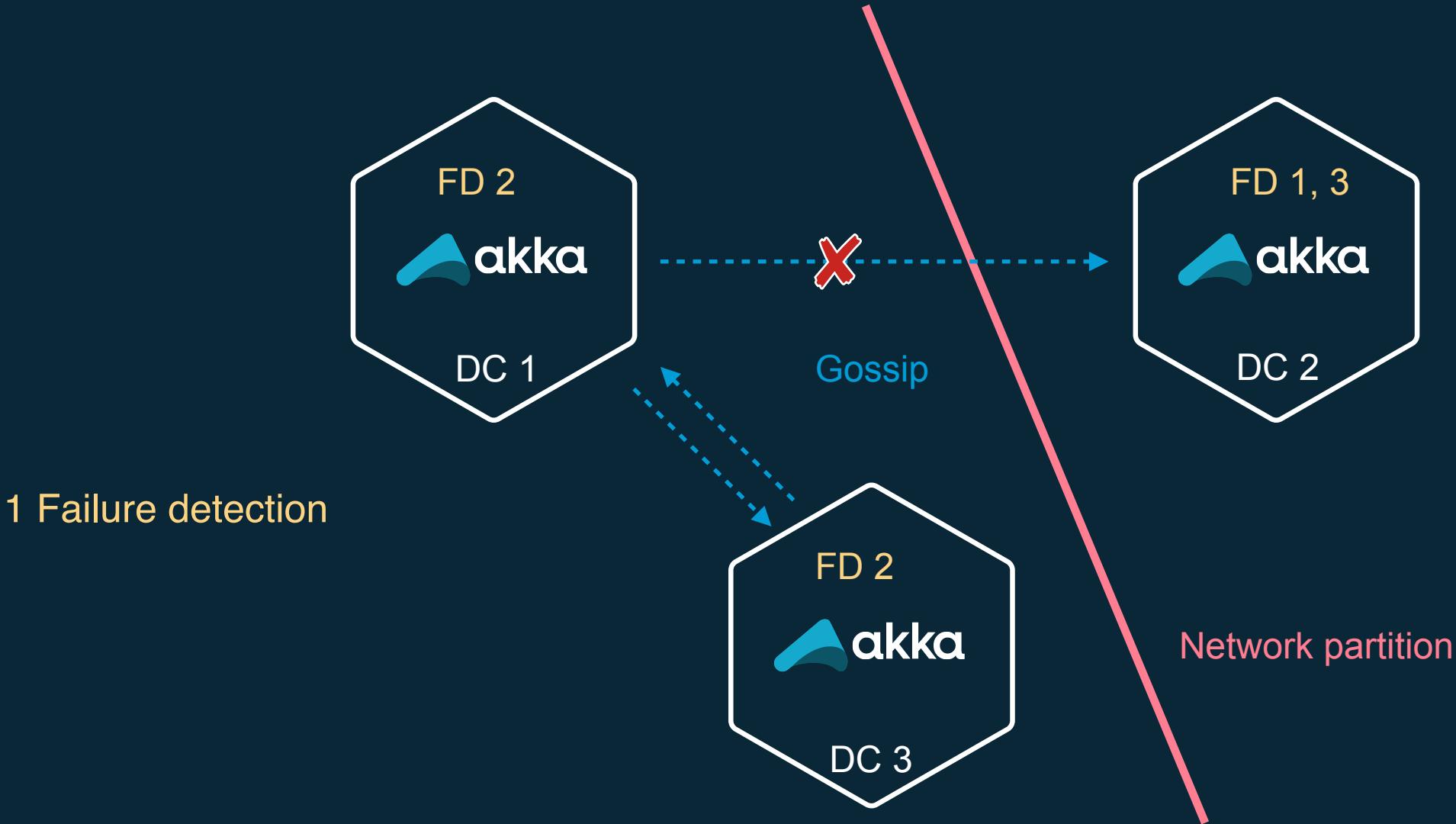
# Handling network partition



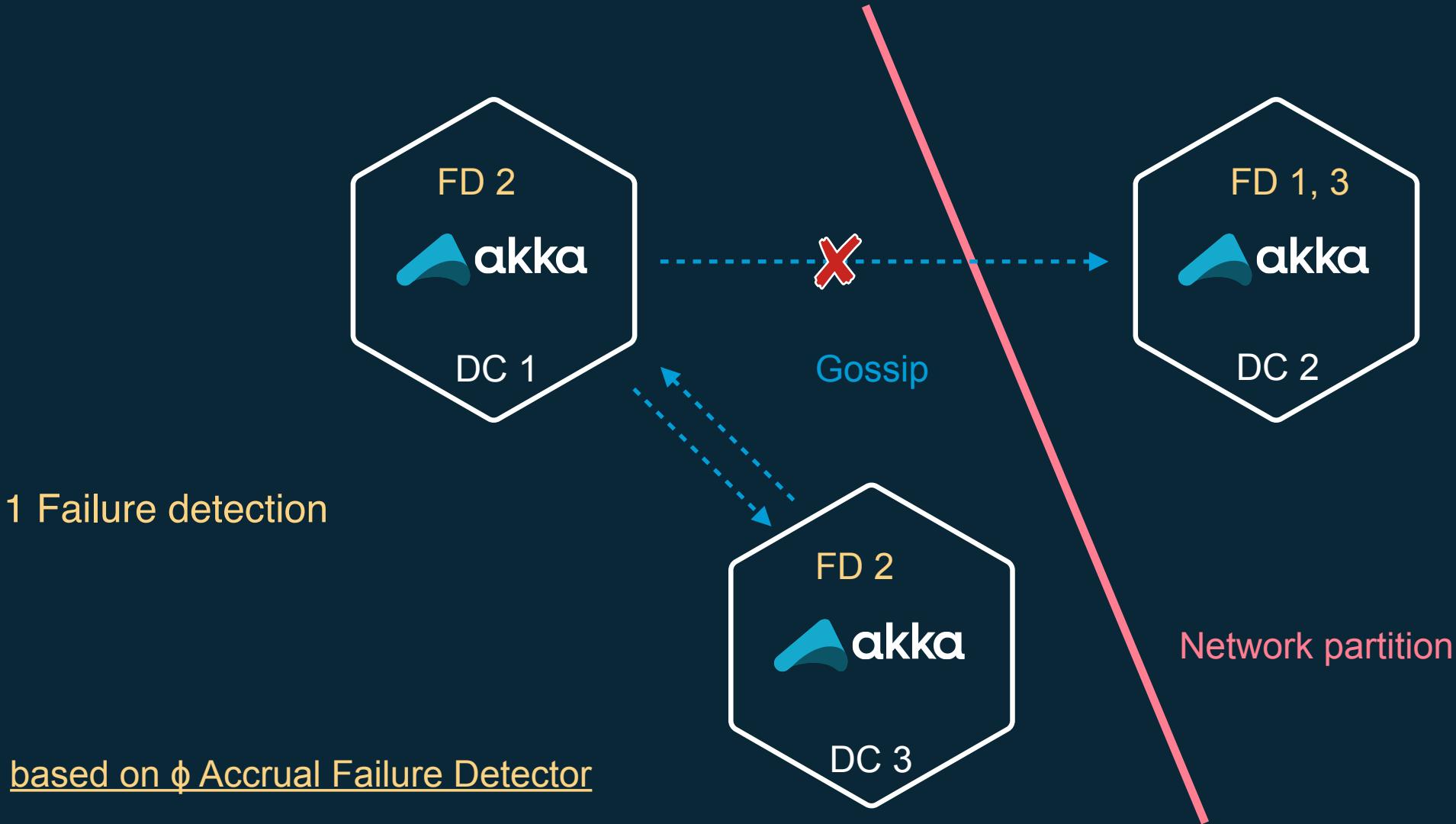
# Handling network partition



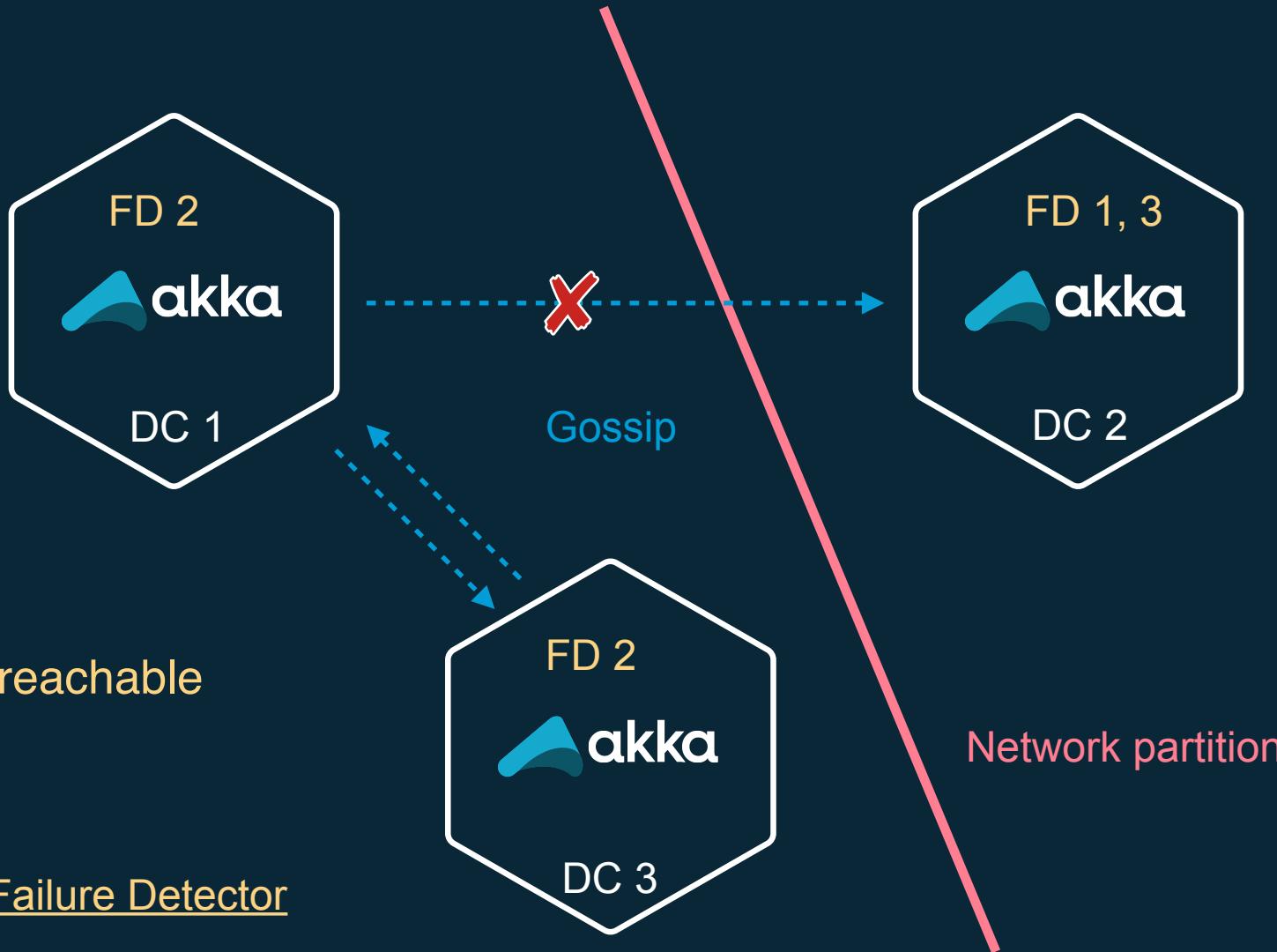
# Handling network partition



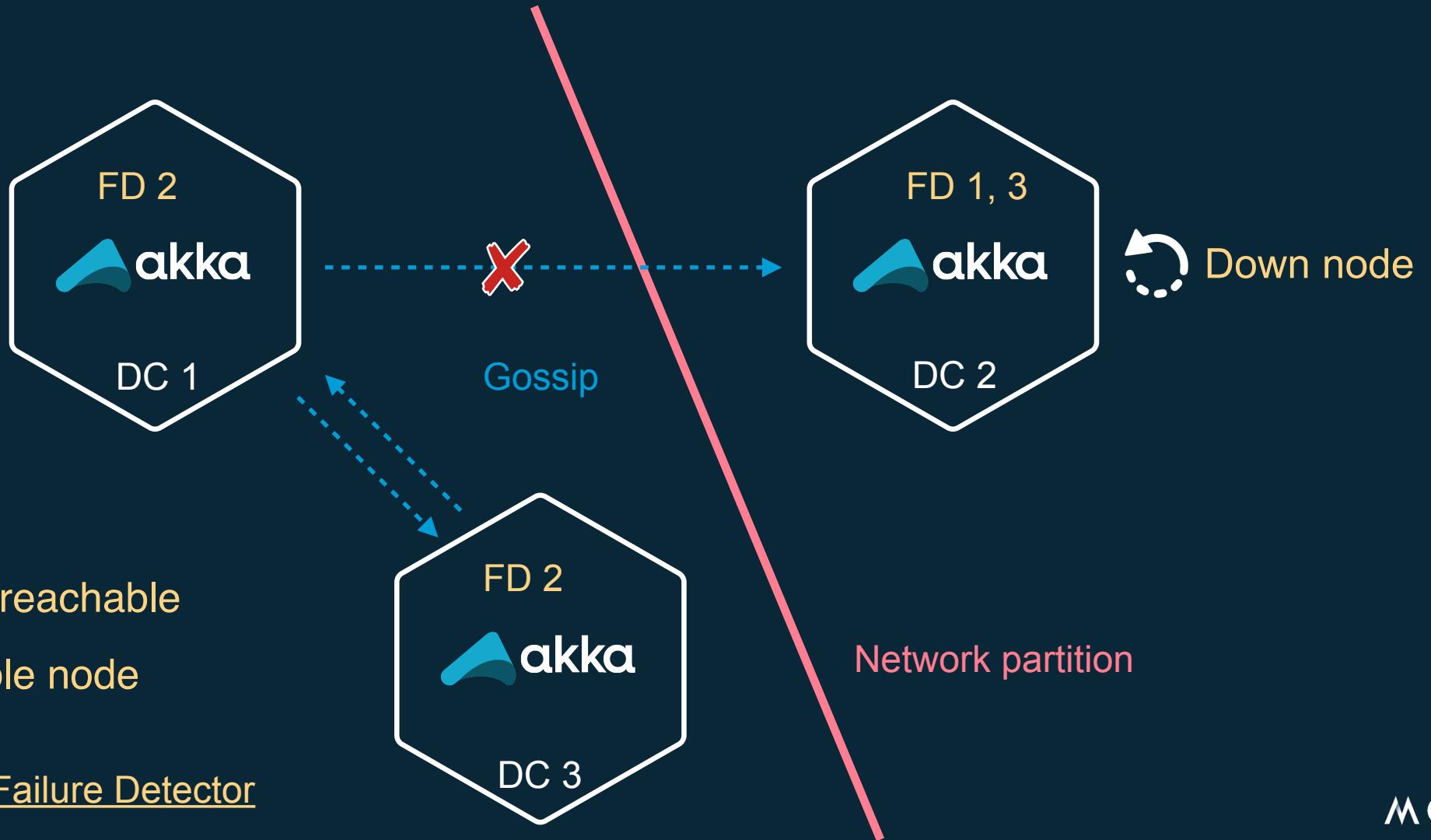
# Handling network partition



# Handling network partition



# Handling network partition



# Strategies

- Static Quorum
- Keep Majority
- Keep Oldest
- Keep Referee

# Libraries

## Commercial

- Lightbend SBR (recommended)

## Open Source

- Akka Cluster Custom Downing
- Akka Down Resolver
- Simple Akka Downing

# Thank you

MOIA

Source code

[github.com/markusjura/akka-cluster-on-kubernetes](https://github.com/markusjura/akka-cluster-on-kubernetes)



 [moia.io/jobs](https://moia.io/jobs)

 [markusjura](#)

 [markusjura](#)