

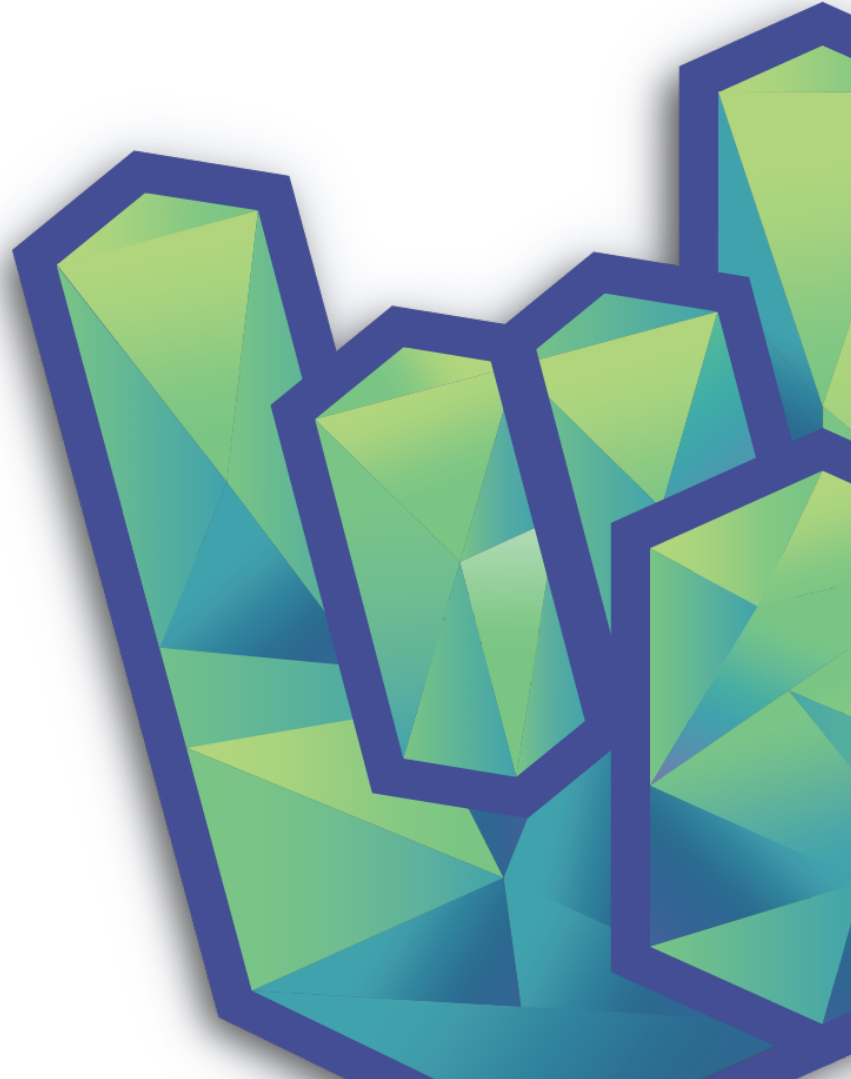
# Integrating with Actors



# Goal

## Make Actors interact with Akka streams

- process elements in streams
- act as a source
- act as a destination



# Recap

## Actors as a flow

```
Flow[Int].ask[String](parallelism = 5)(myActorRef)
```

```
mySource.ask[Int](parallelism = 5)(myActorRef)
```

← equivalent with "via an ask-based flow"

## Actors as a source

```
Source.actorRef[Int](bufferSize = 16, overflowStrategy = OverflowStrategy.dropHead)
```

← materializes to an ActorRef

## Actors as a sink

```
Sink.actorRefWithAck[Int](  
  myActorRef,  
  onInitMessage = MyStreamInit,  
  onCompleteMessage = MyStreamComplete,  
  ackMessage = MyStreamAck,  
  onFailureMessage = ex => MyStreamFail(ex)  
)
```

← "lifecycle" messages

**Akka rocks**

