Microservices with Akka

Akka Http, Akka Persistence and how it fits together.

Agenda

- About me
- Akka Basics (Recap)
- Akka Http
- Akka Persistence

About me

- Christian Börner-Schulte
- Topics
  - Akka
  - Microservices
  - Reactive Systems

Akka Basics (1)

- Toolkit
  - Used in Lagom, Play,…
- Actor Model on JVM
- One Actor System, many Actors
- Everything is an actor

Akka Basics (2)

- Hierarchical Structure
- Supervision and Monitoring
  - Can react on load and failure

Actor Basics (1)

- Computation unit
- Message-Driven communication
- Unique address
  - Never interact with instance directly
- Mailbox
  - Compute one message at a time

Actor Basics (2)

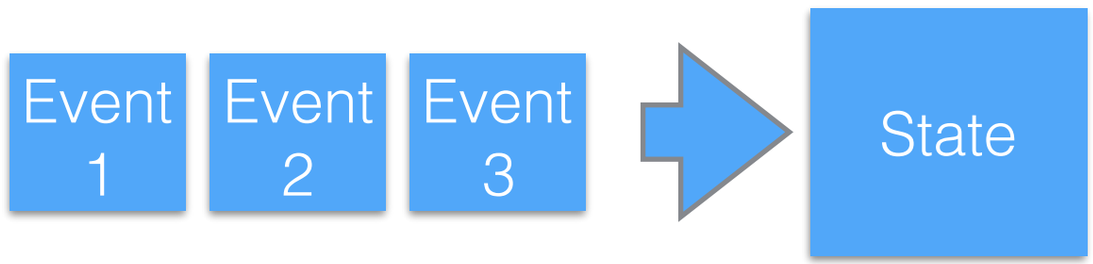
- Receive and react on messages:
- Send messages to (other) actors
- Create other Actors
- Change state/behavior

Akka Http

- Stream based (Akka Streams)
- Containerless
- Build routes hierarchically with directives

Akka Persistence (1)

- Recover State
- Event Sourcing
- Snapshots



- Recovery Process:

    1. Last Snapshot, if present
    2. Events, if present

Akka Persistence (2)

- Suits well for DDD
  - Entity with Value Objects
  - No Reposistory-Service
- CQRS-ish

Further steps

- Akka Cluster Sharding

- Multiple JVMs

THX

- github: daesul

- Email: christian.boerner.schulte@gmail.com