

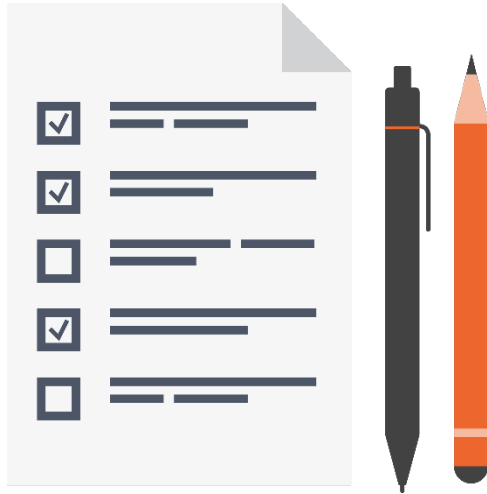
# Creating Actor Hierarchies and Isolating Faults



Jason Roberts

@robertsjason | dontcodetired.com

# Overview



Creating movie streaming actor hierarchy

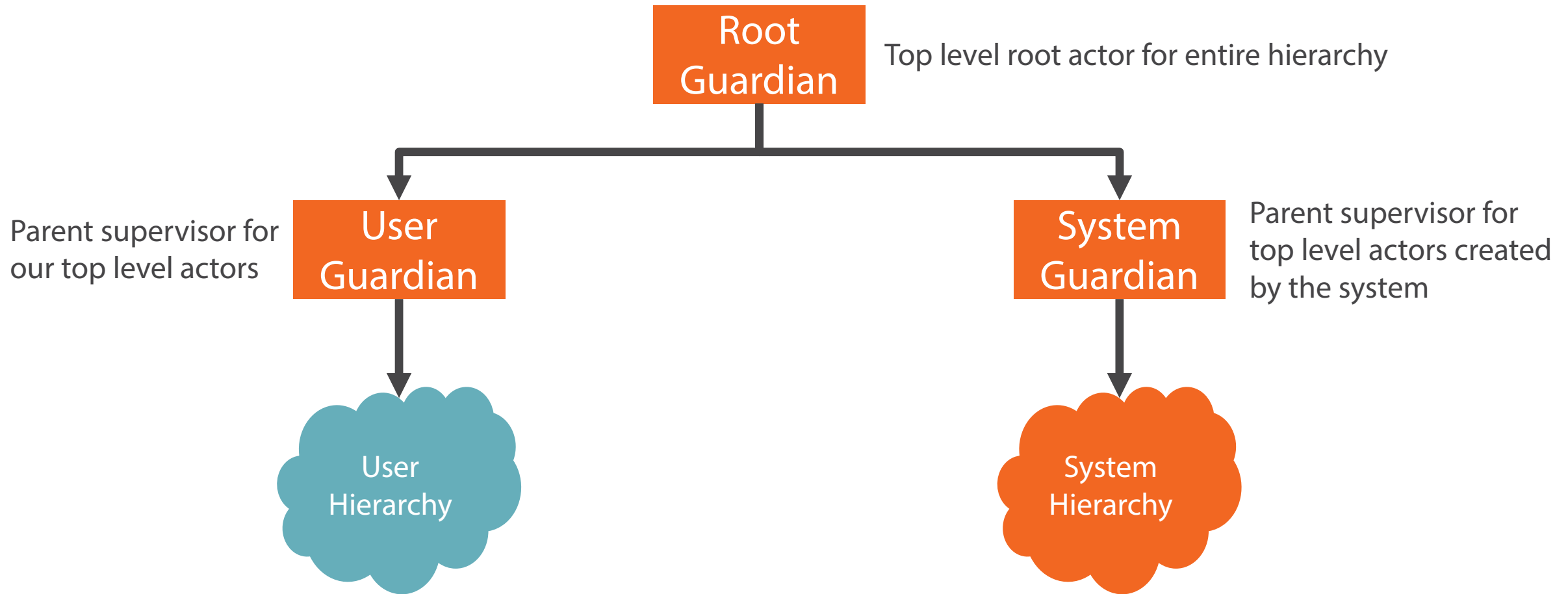
System and top level actors

Actor paths and actor selections

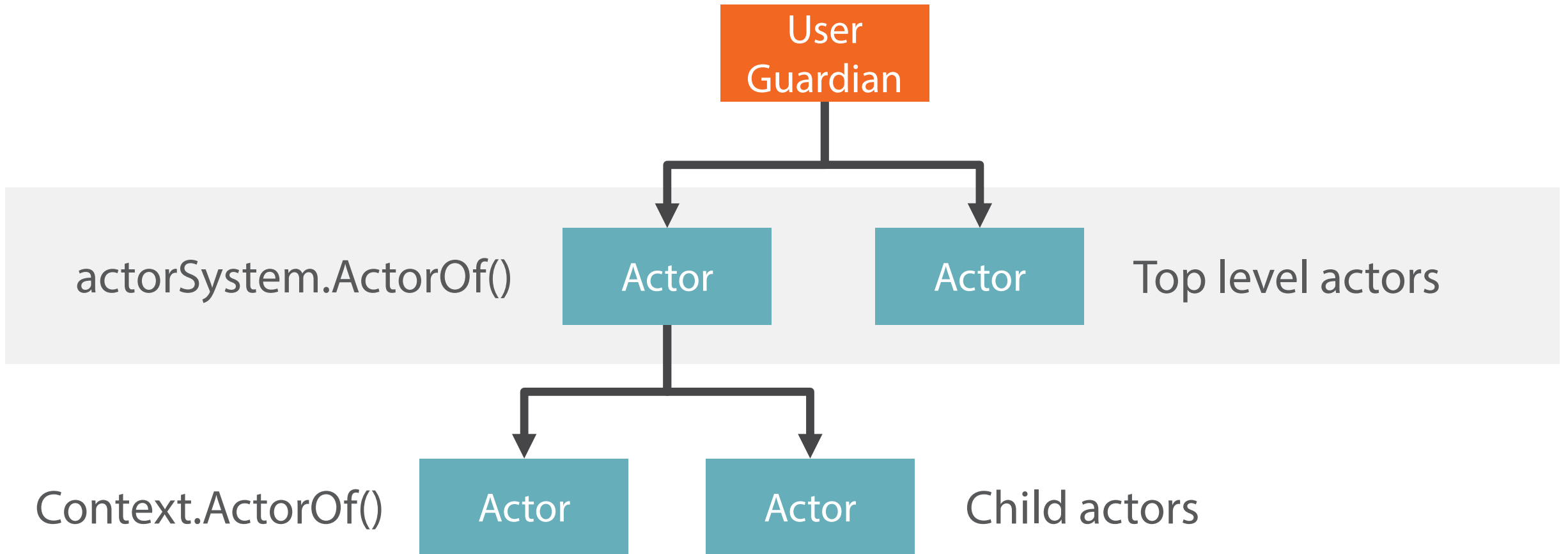
Default parental supervision strategy

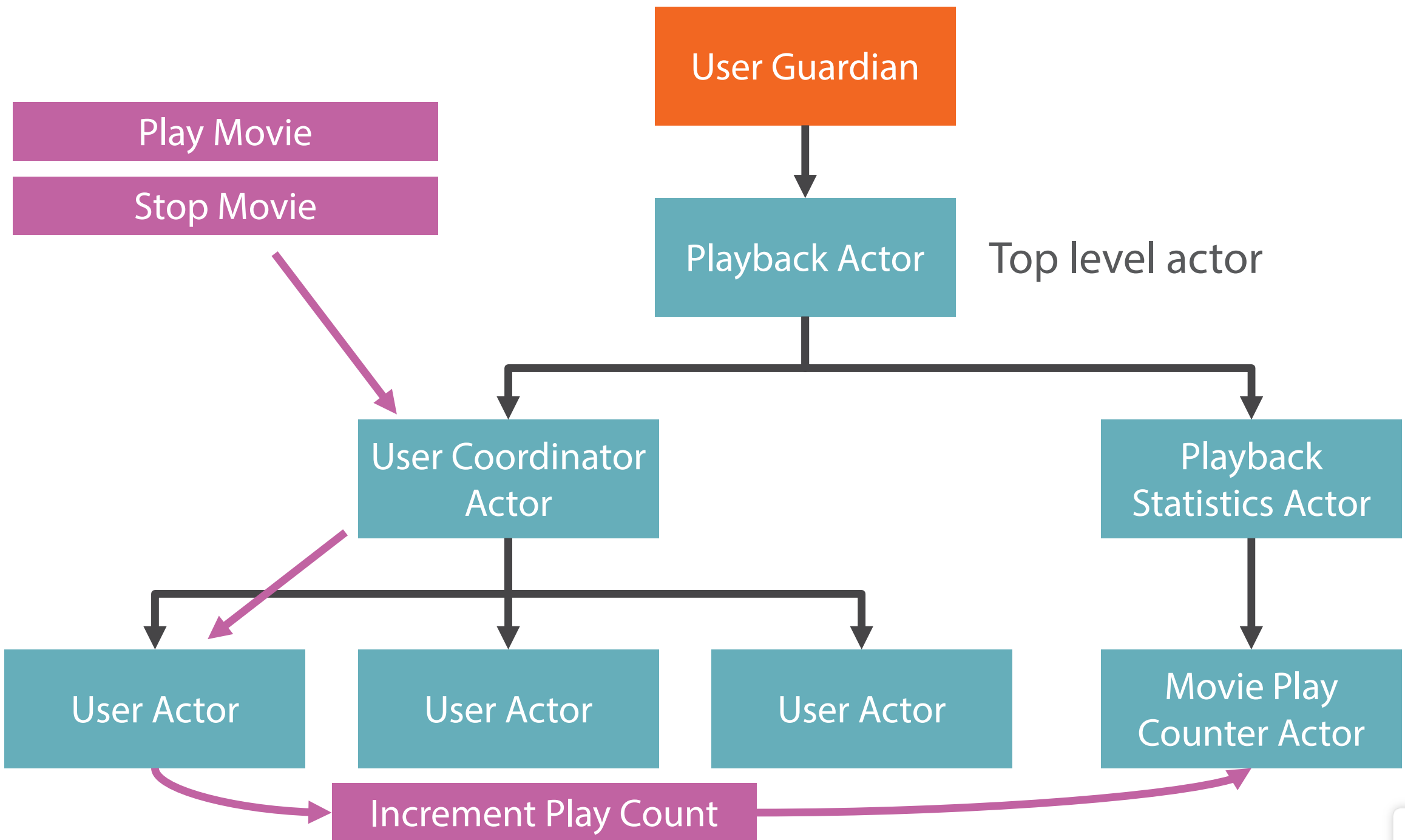
Creating a custom parental supervision strategy

# The Actor Supervision Hierarchy



# The Actor Supervision Hierarchy



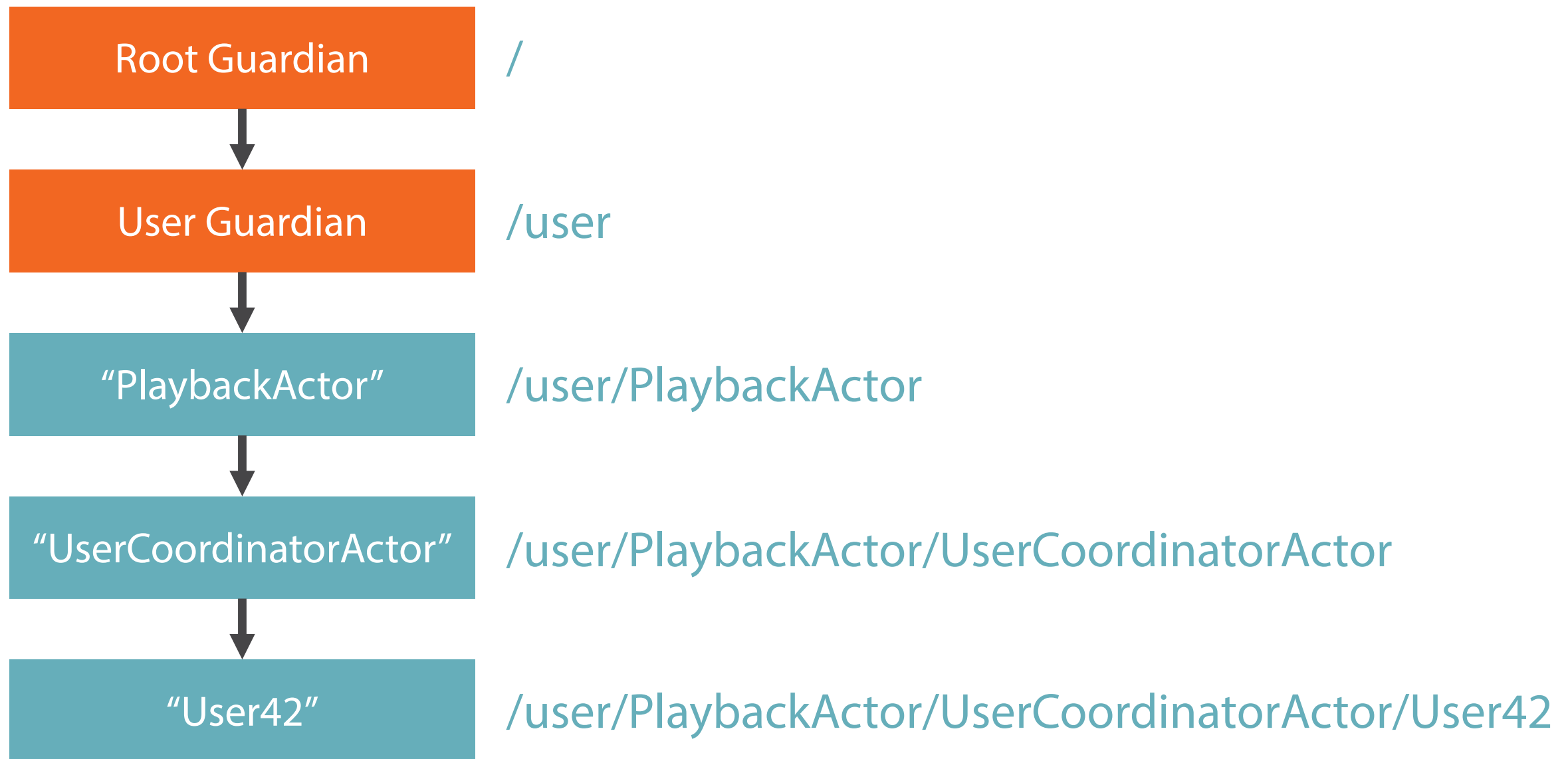


# Actor Paths

Protocol	- single process or inter-process transport protocol
Actor System	- name of the actor system instance
Address	- remote actor system IP address / port
Hierarchy Path	- path to actor in the hierarchy

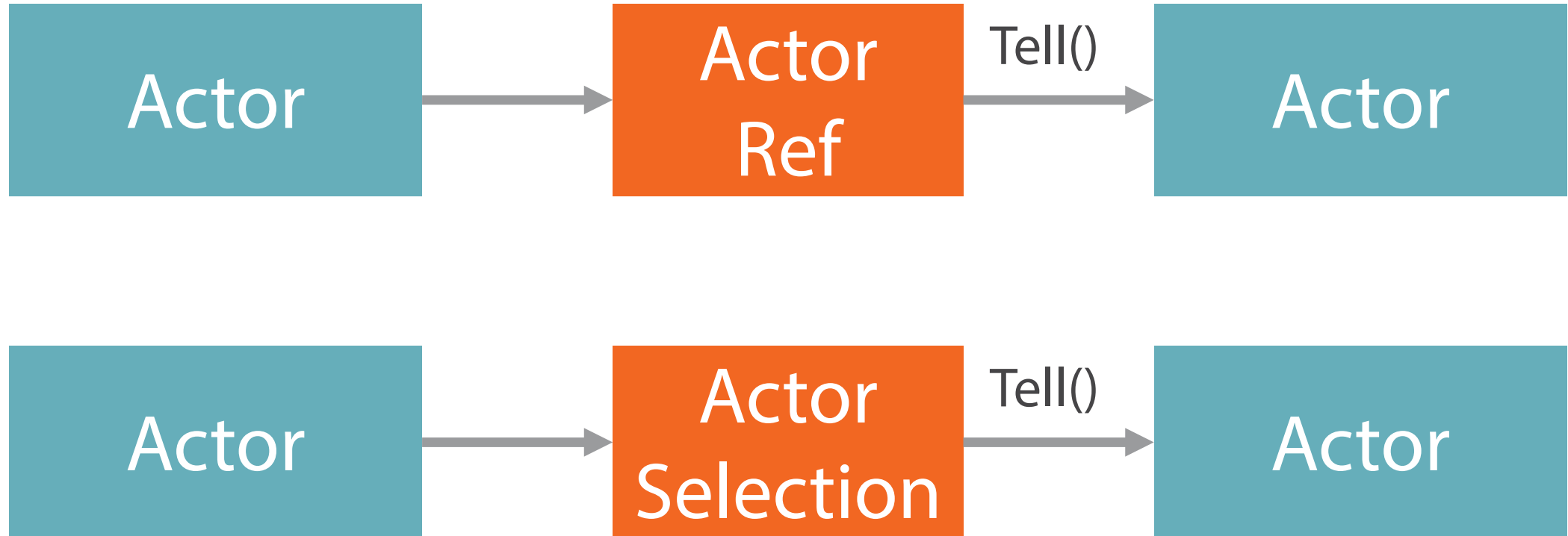
akka://MovieStreamingActorSystem/user/Playback

akka.tcp://MovieStreamingActorSystem@122.241.30.4:9002/user/Playback



akka://MovieStreamingActorSystem/user/PlaybackActor/UserCoordinatorActor/User42

# Actor Selection





# Actor Selection

```
var message = new PlayMovieMessage(movieTitle, userId);
```

```
MovieStreamingActorSystem
```

```
.ActorSelection(  
    "akka://MovieStreamingActorSystem/user/Playback/UserCoordinator")
```

```
.Tell(message);
```

# Actor Selection

```
var message = new PlayMovieMessage(movieTitle, userId);
```

```
MovieStreamingActorSystem  
    .ActorSelection("/user/Playback/UserCoordinator")  
  
    .Tell(message);
```

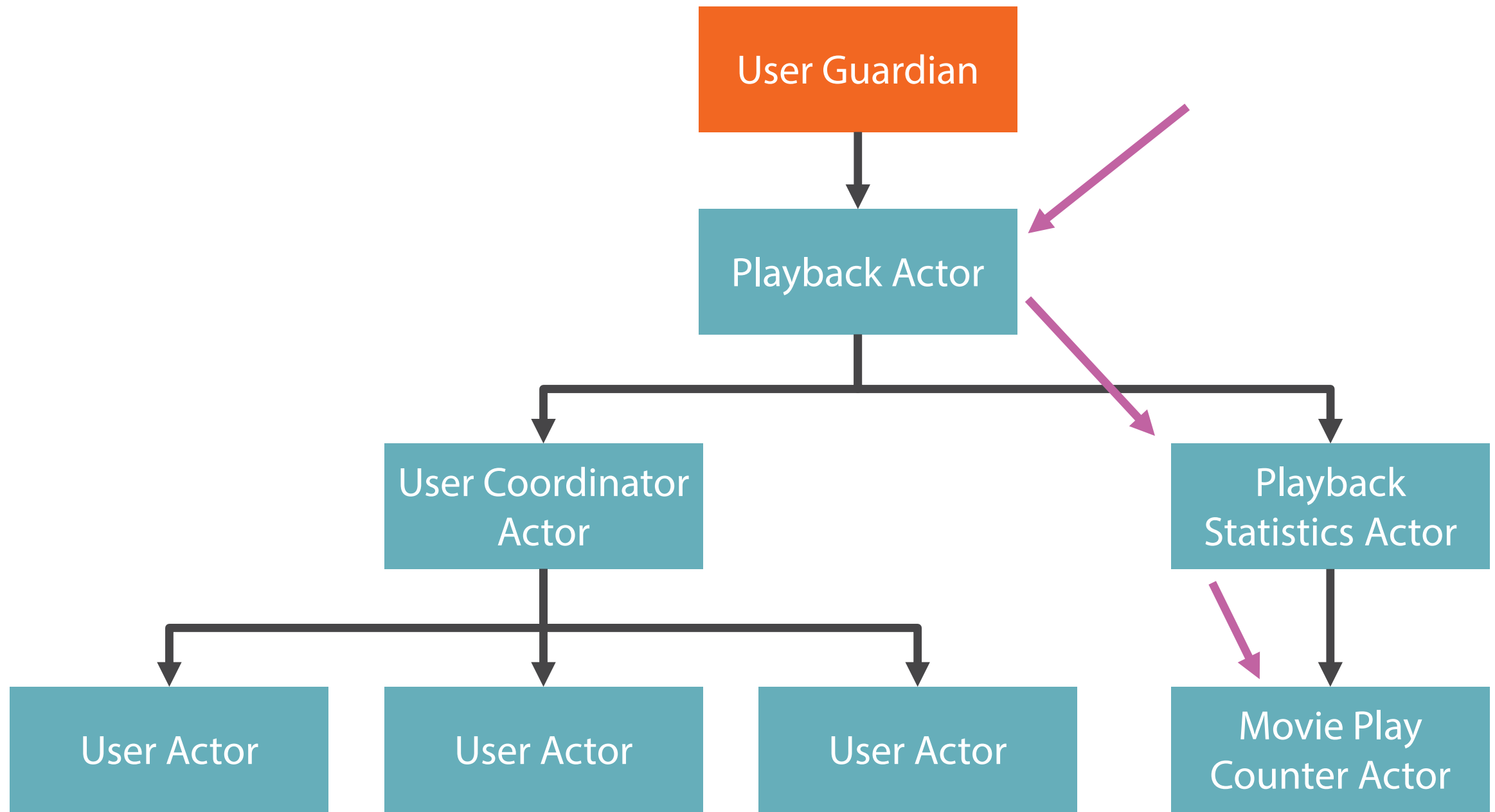
```
Context.ActorSelection(  
    "/user/Playback/PlaybackStatistics/MoviePlayCounter")
```

```
Context.ActorSelection(  
    "../..PlaybackStatistics/MoviePlayCounter")
```

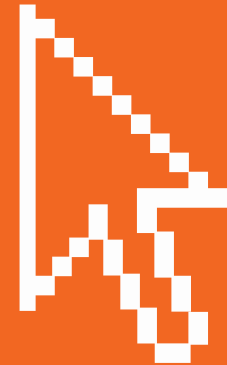
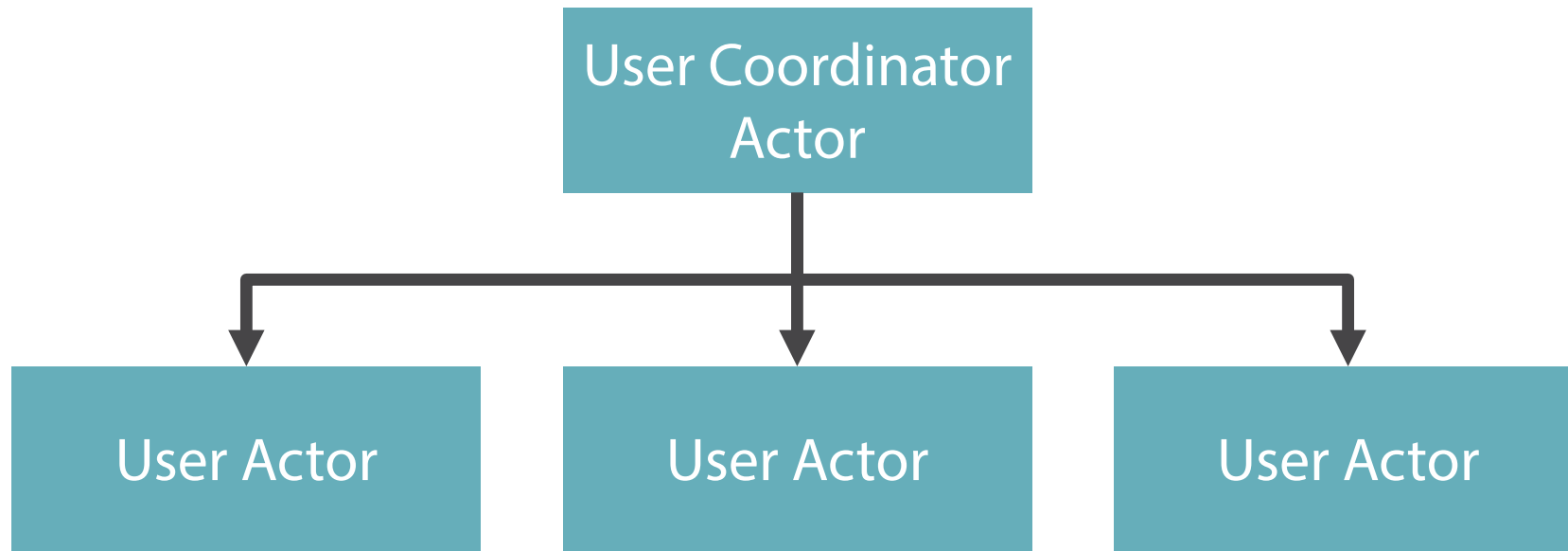
## Relative Actor Selection Paths

Within actor code, we can use relative paths

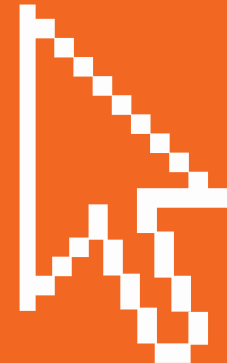
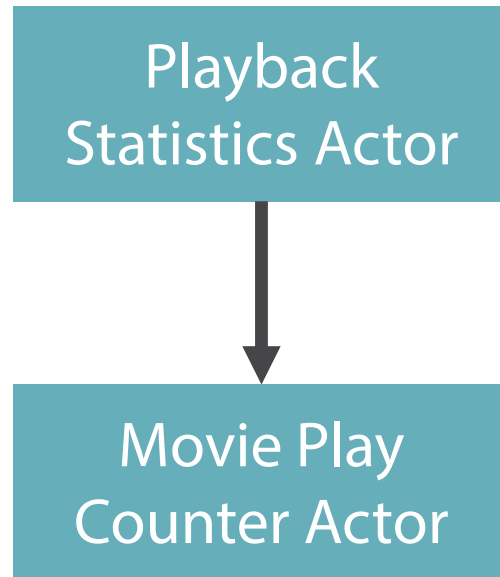
Supervision hierarchies do not  
define message passing routes



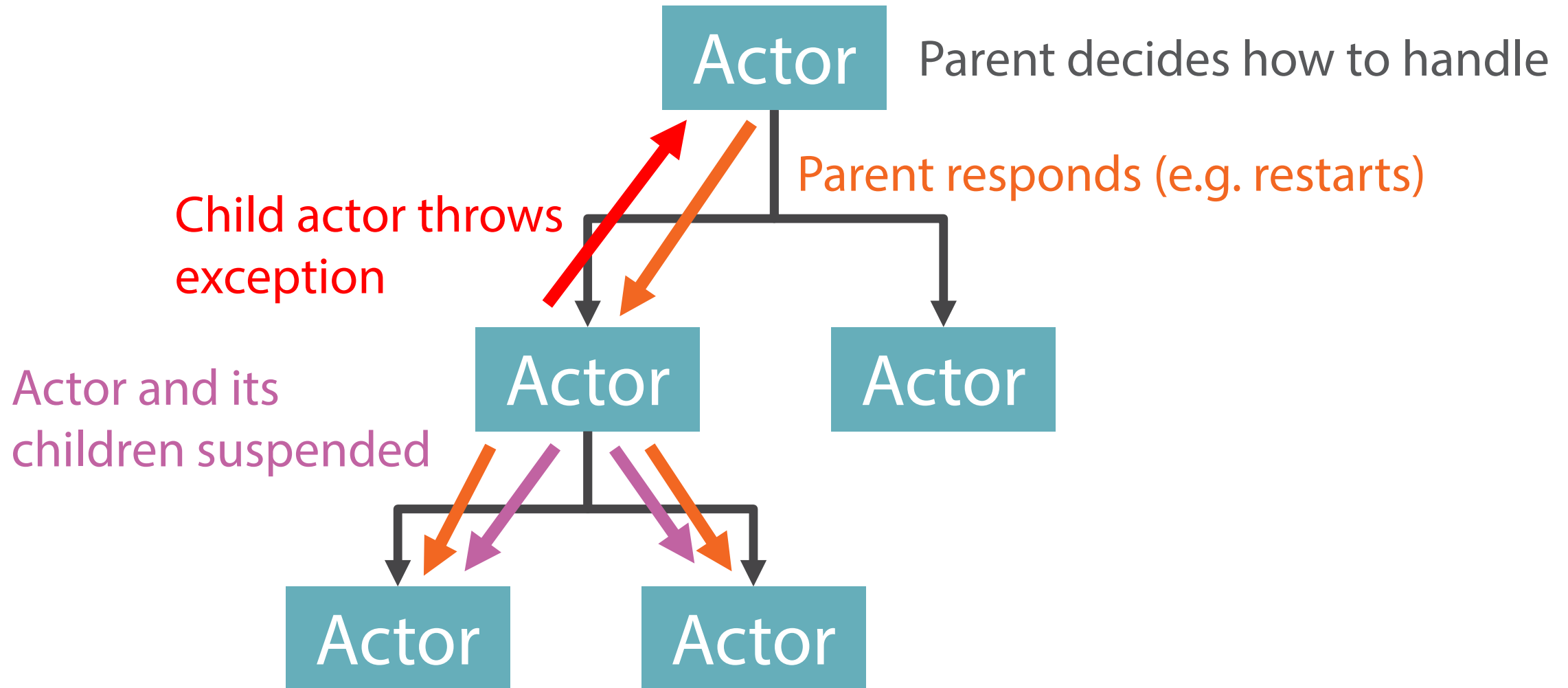
# Creating the UserCoordinator Actor



# Creating the MoviePlayCounter Actor



# Parental Supervision Choices





# Parental Supervision Choices



Resume



Restart



Stop



Escalate

# Parental Supervision Choices



Exception is ignored

Child actor state is maintained

Child actor's children are also resumed

# Parental Supervision Choices



Child actor is restarted

Child actor state is lost

Child actor's children also restarted by default

Default behaviour if no custom strategy is implemented

Remaining mailbox messages are preserved

# Parental Supervision Choices



Child actor permanently terminated

Child actor's children also permanently terminated

# Parental Supervision Choices



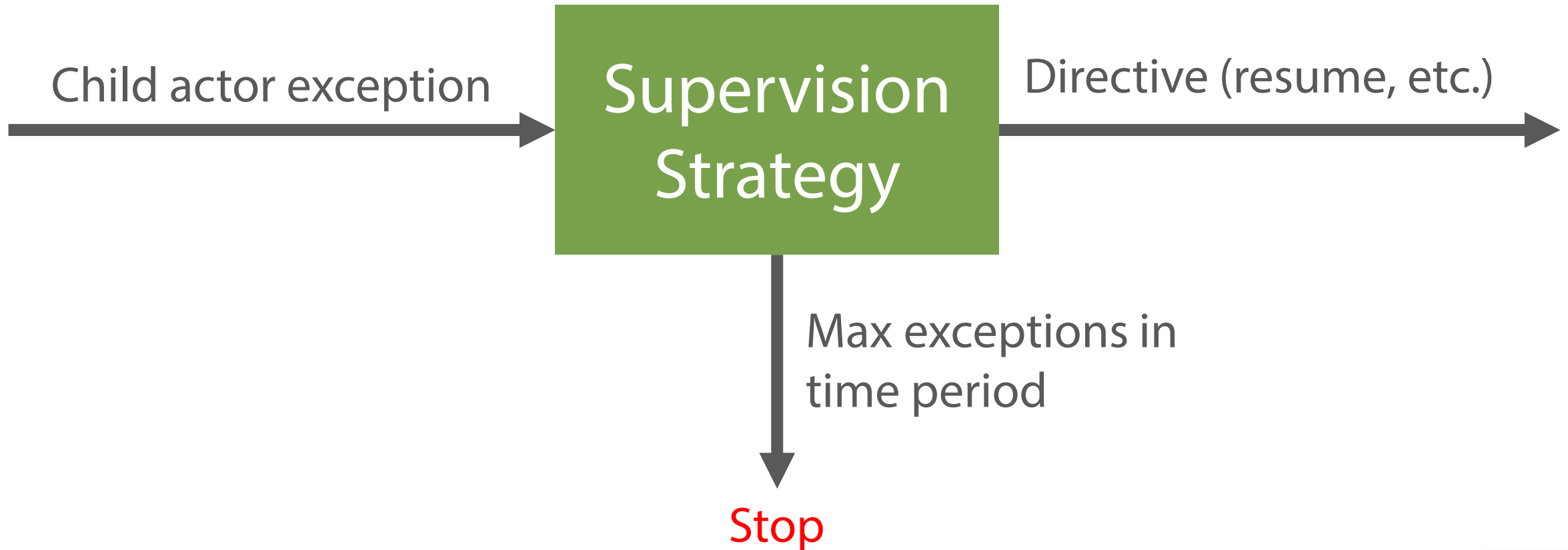
Exception escalated to the parent actor's parent

Parent actor itself also becomes suspended

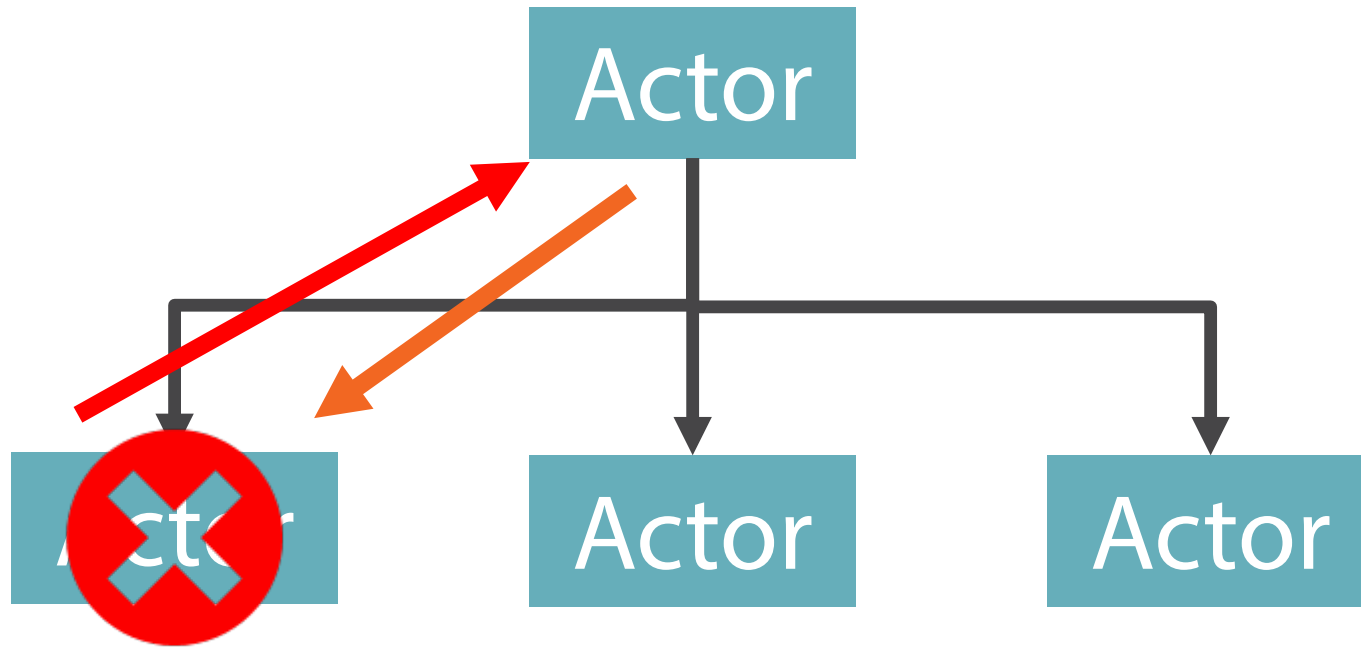
All the other parent's children also become suspended

# Akka.Persistence

# Supervision Strategies



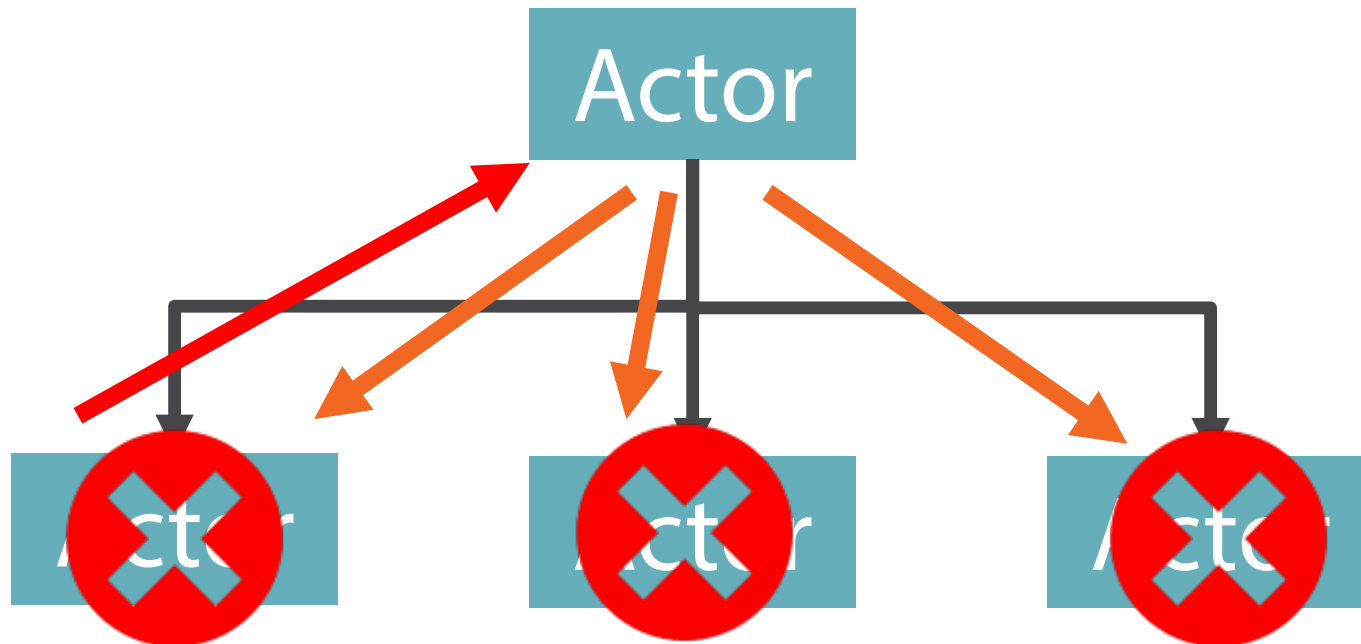
# Default OneForOneStrategy



Directive (e.g. stop)  
applied only to  
failing child



# AllForOneStrategy

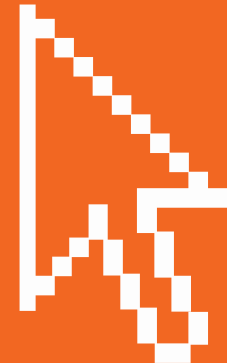


Directive (e.g. stop)  
applied to **all** children

# Default Supervisor Strategy



# Custom Supervisor Strategy



# Summary



Creating movie streaming actor hierarchy

User guardian and top level actors

`Context.ActorSelection("/user/...")`

Resume, restart, stop, escalate

Default parental supervision strategy

Override SupervisorStrategy

`new OneForOneStrategy(...)`

Next:

# Deploying and Messaging Remote Actors