

Thinking in Events and Commands



Jason Roberts

.NET MVP

@robertsjason dontcodetired.com



Overview



Understanding events and commands

Persist events

Change messages into events/commands

Refactor PlayerCoordinatorActor

Refactor PlayerActor



Understanding Events and Commands

Commands

Not persisted

Request that actor take some action

May result in (0,1,m) events created

May be validated/“rejected”

Naming convention: verb (imperative)

E.g. HitPlayer

Events

Persisted

Something that took place

Represent a change of state happened

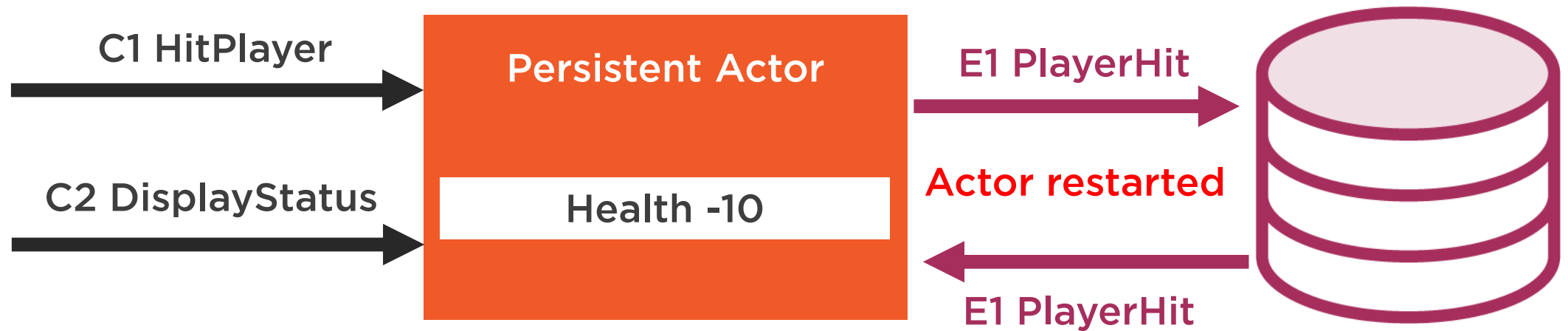
Replayed on recovery

Naming convention: verb (past)

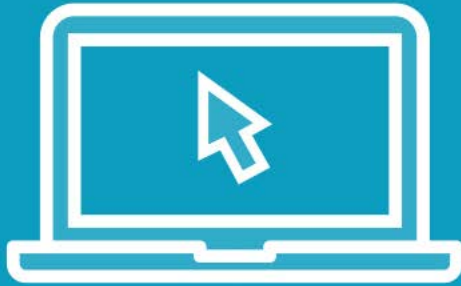
E.g. PlayerHit



Understanding Events and Commands



Demo



Add Event and Command folders

Messages → Events and Commands

Refactor PlayerCoordinatorActor

Refactor PlayerActor

Delete SQL server journal entries

Run app

Restart app



Summary



Understanding events and commands

Commands → 0,1,m events

Events, not commands persisted

Changed messages into events/commands

Refactored PlayerCoordinatorActor

Refactored PlayerActor



Next:

Improving Actor Recovery Time
with Snapshots

