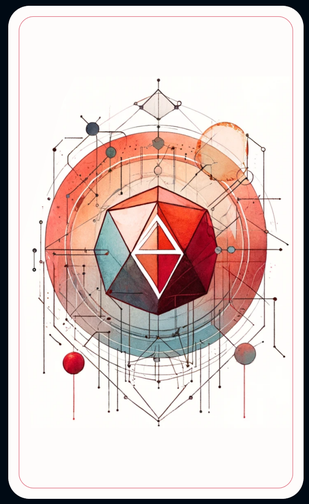
v18 is now available

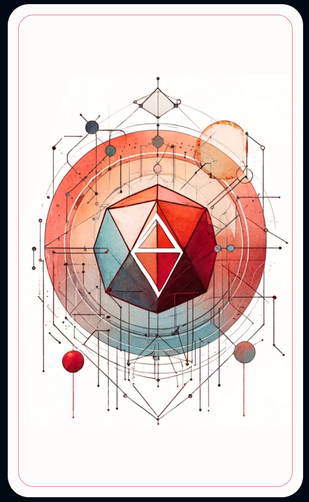
"How do we architect with Signals?"

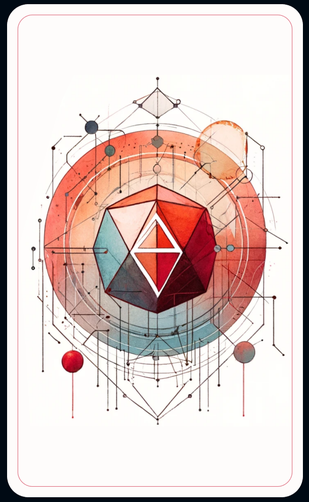
Let's do a Tarot reading

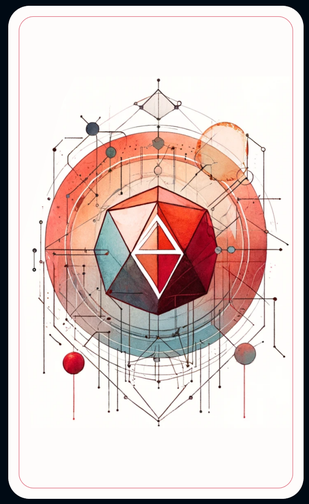
Let's do a Tarot reading

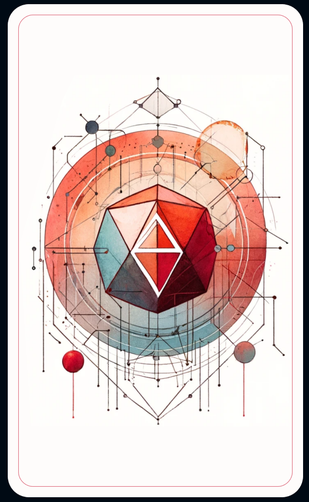
Let's do a Tarot reading

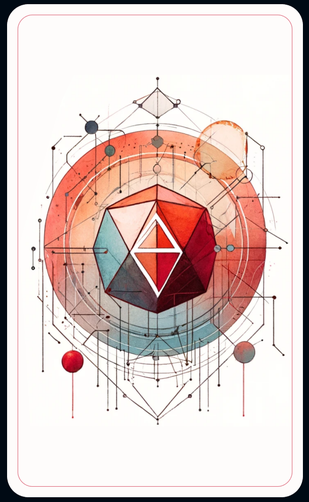


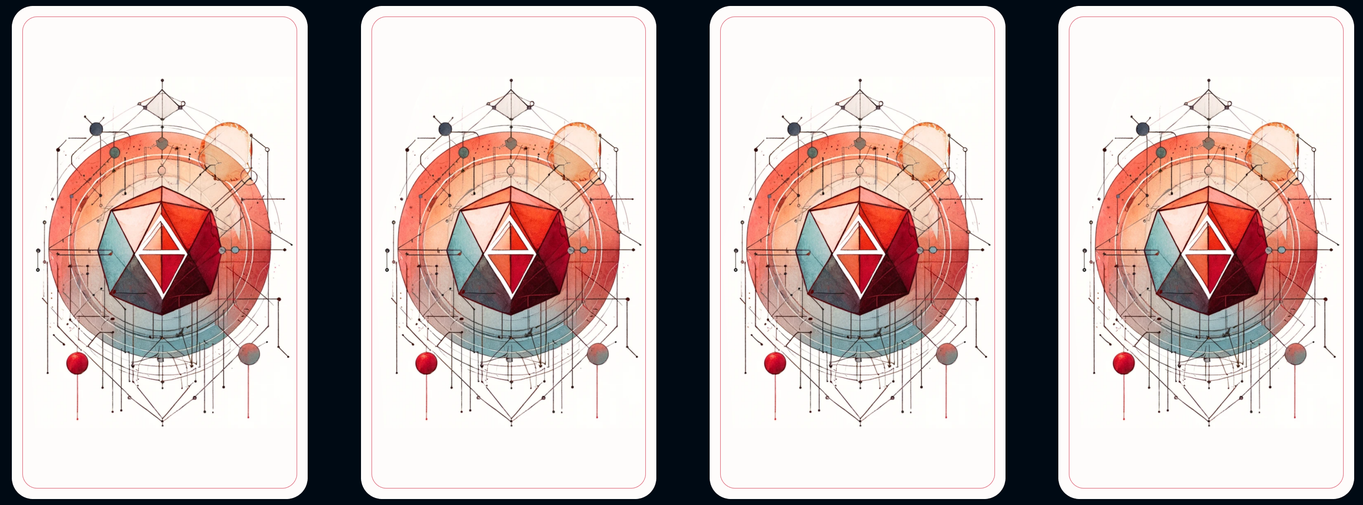


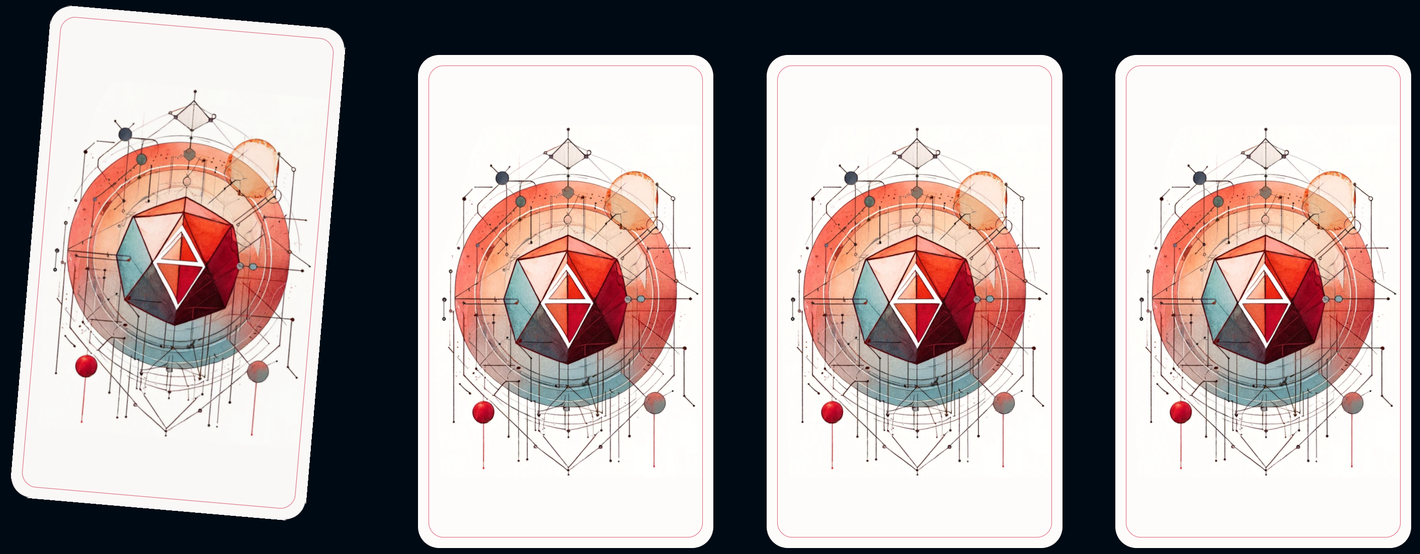












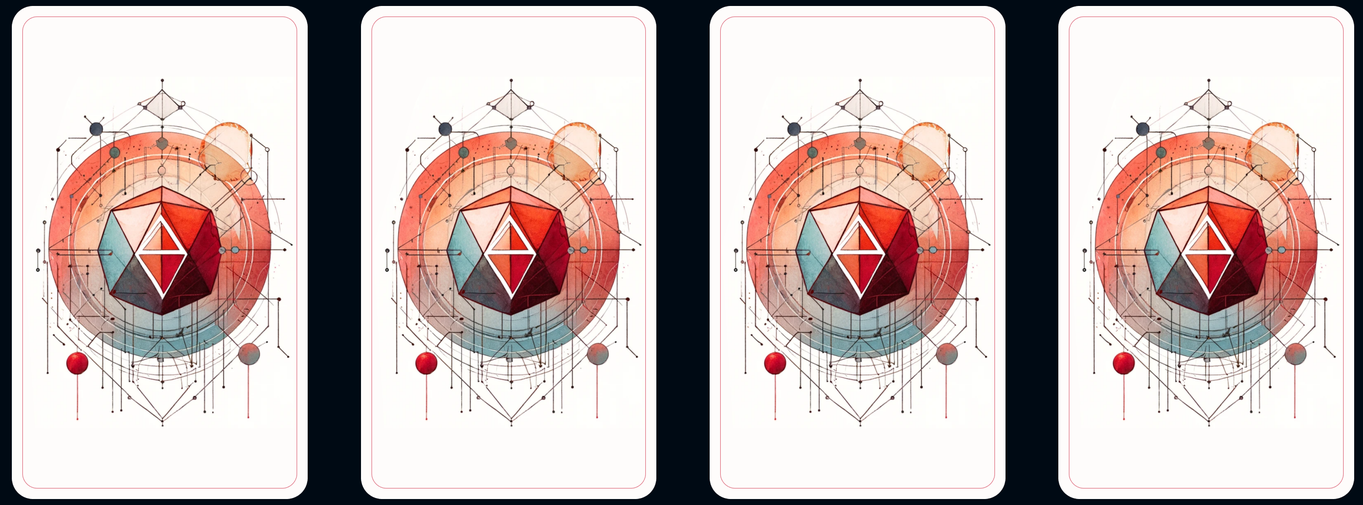
Mike Ryan

NgRx Co-Creator & Team Alumn

GDE in **Angular**

Consultant at **LiveLoveApp**

Building **Polaris**



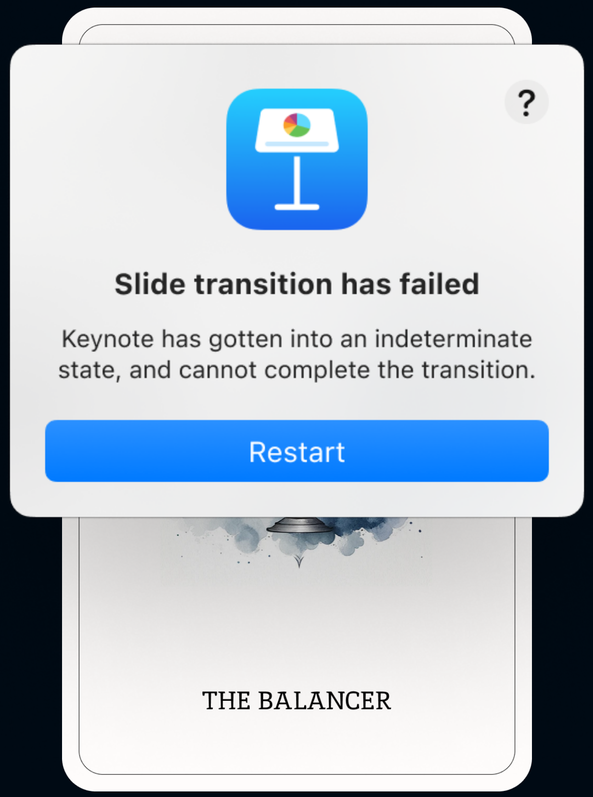

THE BALANCER


◇

THE BALANCER

◇

THE BALANCER


?
Slide transition has failed
Keynote has gotten into an indeterminate state, and cannot complete the transition.
Restart
THE BALANCER

*** Kernel Panic ***

An unexpected error was detected. The system has encountered a fatal error caused by an application in an unstable state.
Immediate action is required to prevent further damage.

*** Kernel Extensions in backtrace:
com.apple.driver.AppleKeynote (1.0.0) <3FAE6F1D-5D3F-4D93-8C79-EC4A66B3B4F2>
    dependency: com.apple.driver.CoreGraphics
    dependency: com.apple.driver.QuartzCore
    dependency: com.apple.iokit.IOGraphicsFamily

*** Panic Report ***
panic(cpu 0 caller 0xffffff800bca3e5a): **"Keynote has entered a bad state due to excessive slide transitions and animations.
System stability cannot be guaranteed. Please calm down on the animations, you're not presenting a Marvel movie."**

Debugger message: panic
Memory ID: 0xff000f
OS version: 20.6.0 (Build 20G224)
Kernel version: Darwin Kernel Version 20.6.0: Thu Sep 9 19:22:28 PDT 2021; root:xnu-7195.141.2~5/RELEASE_X86_64
Kernel UUID: 9D6E1EBD-77A7-3CB7-9368-7A35EF1D1A28
Kernel slide: 0x000000000b000000
Kernel text base: 0xffffff800b200000
__HIB text base: 0xffffff800b100000
System model name: MacBookPro16,1 (Mac-CFF7D910A743CAAF)

System uptime in nanoseconds: 13248038012358
last loaded kext at 133304722385: com.apple.filesystems.msdosfs 1.10 (addr 0xffffff7fa3d42000, size 69632)
last unloaded kext at 132904637682: com.apple.driver.AppleIntelMCEReporter 115 (addr 0xffffff7fa1b84000, size 49152)

Loaded kexts:
com.apple.driver.AppleKeynote   1.0.0
com.apple.driver.CoreGraphics   1.0.0
com.apple.driver.QuartzCore   1.0.0
com.apple.iokit.IOGraphicsFamily   1.0.0
<Other loaded kexts>

BSD process name corresponding to current thread: Keynote
Boot args: -v keepsyms=1 dart=0

Mac OS version:

"Turn it off and on again"

"Clear the cache"

"Refresh the page"

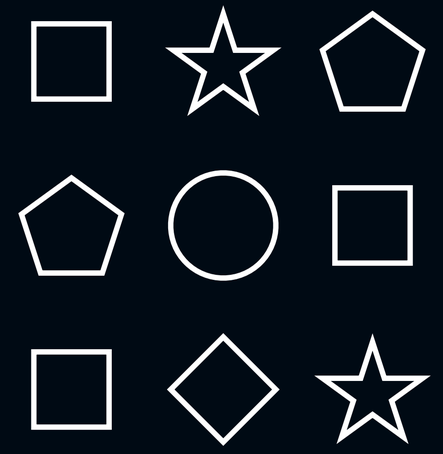
State is hard

Out of the Tarpit

“Complexity is the single major difficulty in the successful development of large-scale software”

Out of the Tarpit

“Major contributor to this complexity is primarily **state** along with code volume and **complex control flow**”




f( )
=>


f( ) =>


f( )
=>

Demo

Architect your **state** with intentionality

Architect your

h intentionality

Architect your **state** with intentionality

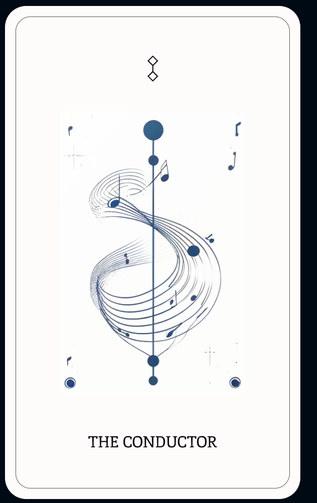
◇

THE BALANCER


THE BALANCER



THE BALANCER


THE CONDUCTOR

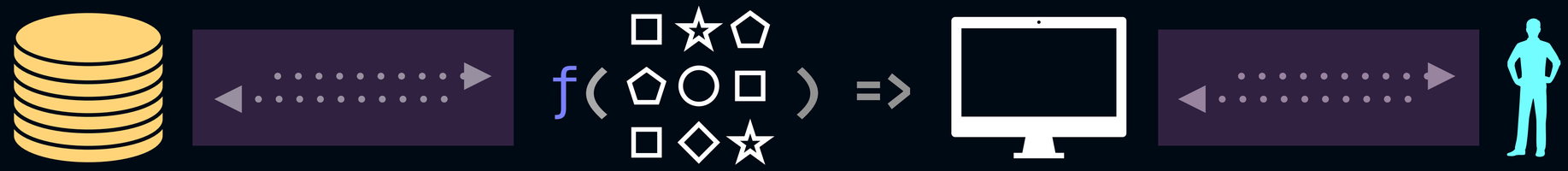
THE CONDUCTOR

Take Signals seriously

```
provideExperimentalZonelessChangeDetection()
```


f( )
=>

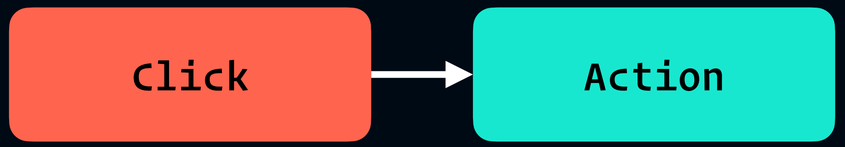
Zones monitor sources of **potential** change



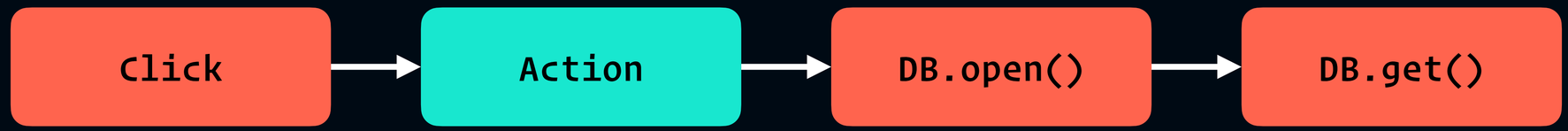
Signals monitor **actual** change

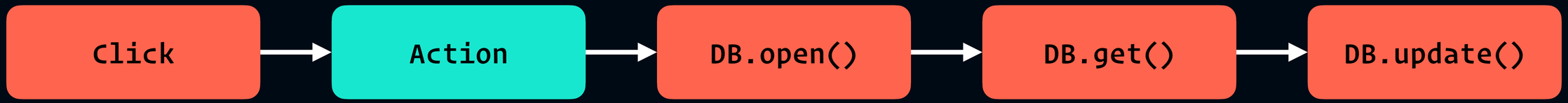
Demo

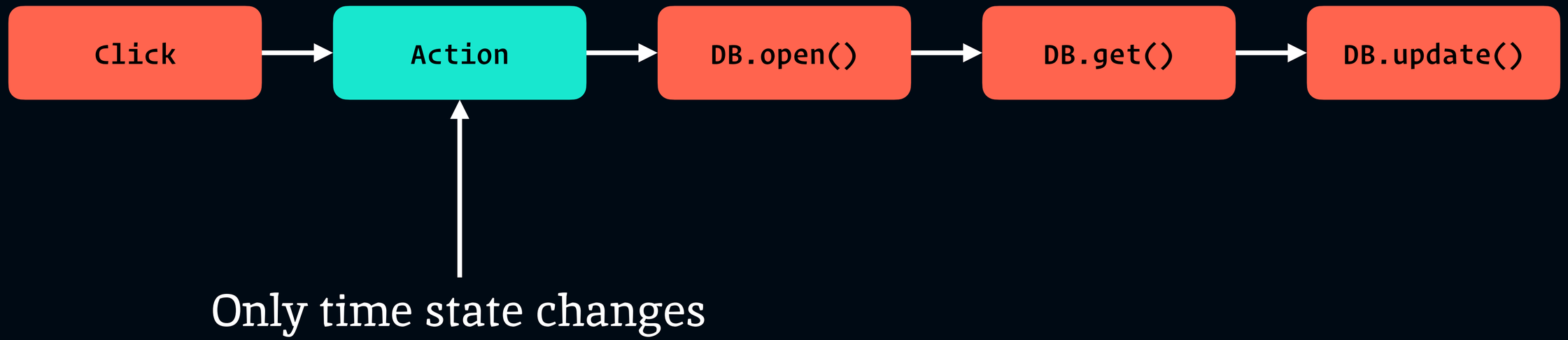

Click

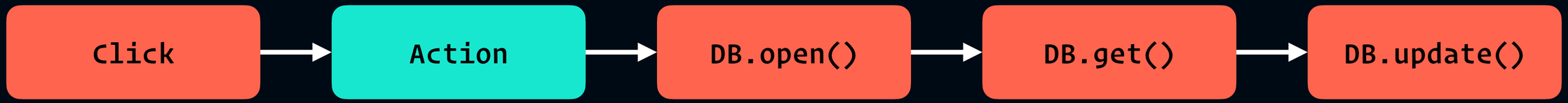

Click
Action


Click
Action
DB.open()


Click
Action
DB.open()
DB.get()

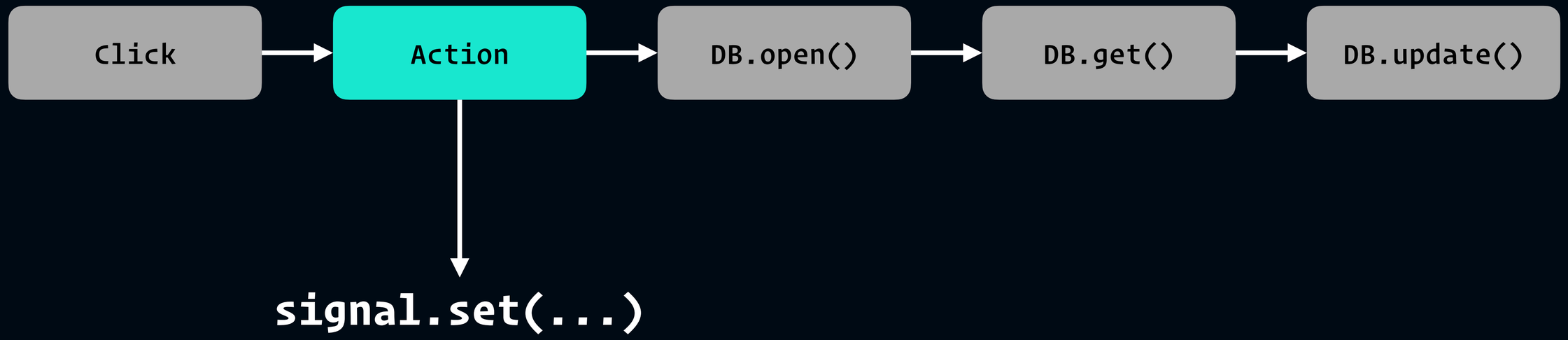

Click
Action
DB.open()
DB.get()
DB.update()


Click
Action
DB.open()
DB.get()
DB.update()
Only time state changes


Click
Action
DB.open()
DB.get()
DB.update()

ZoneJS does not know which events caused a state change

Demo


Click
Action
DB.open()
DB.get()
DB.update()
signal.set(...)

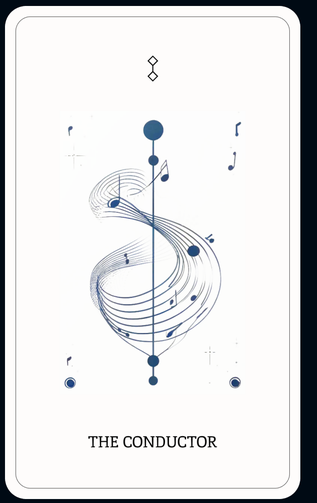
Stack traces!

```
Error: Something went wrong
    at Object.throwError (main.js:15)
    at ZoneDelegate.invoke (zone-evergreen.js:364)
    at Zone.run (zone-evergreen.js:134)
    at NgZone.run (core.js:41892)
    at AppComponent.throwError (app.component.ts:10)
    at HTMLButtonElement.<anonymous> (app.component.html:2)
    at ZoneDelegate.invokeTask (zone-evergreen.js:399)
    at Zone.runTask (zone-evergreen.js:167)
    at ZoneTask.invokeTask [as invoke] (zone-evergreen.js:480)
    at ZoneTask.invoke (zone-evergreen.js:469)
    at timer (zone-evergreen.js:2552)
    at ZoneDelegate.invokeTask (zone-evergreen.js:400)
    at Object.onInvokeTask (core.js:39900)
    at ZoneDelegate.invokeTask (zone-evergreen.js:399)
    at Zone.runTask (zone-evergreen.js:167)
    at invokeTask (zone-evergreen.js:519)
    at ZoneTask.invoke (zone-evergreen.js:508)
    at timer (zone-evergreen.js:2552)
```

```
Error: Something went wrong
    at throwError (main.js:15)
    at AppComponent.throwError (app.component.ts:10)
    at HTMLButtonElement.<anonymous> (app.component.html:2)
    at HTMLButtonElement.onclick (app.component.html:2)
```

THE CONDUCTOR


THE BALANCER


THE CONDUCTOR

◇

THE BALANCER

◇
◇

THE CONDUCTOR

◇◇◇

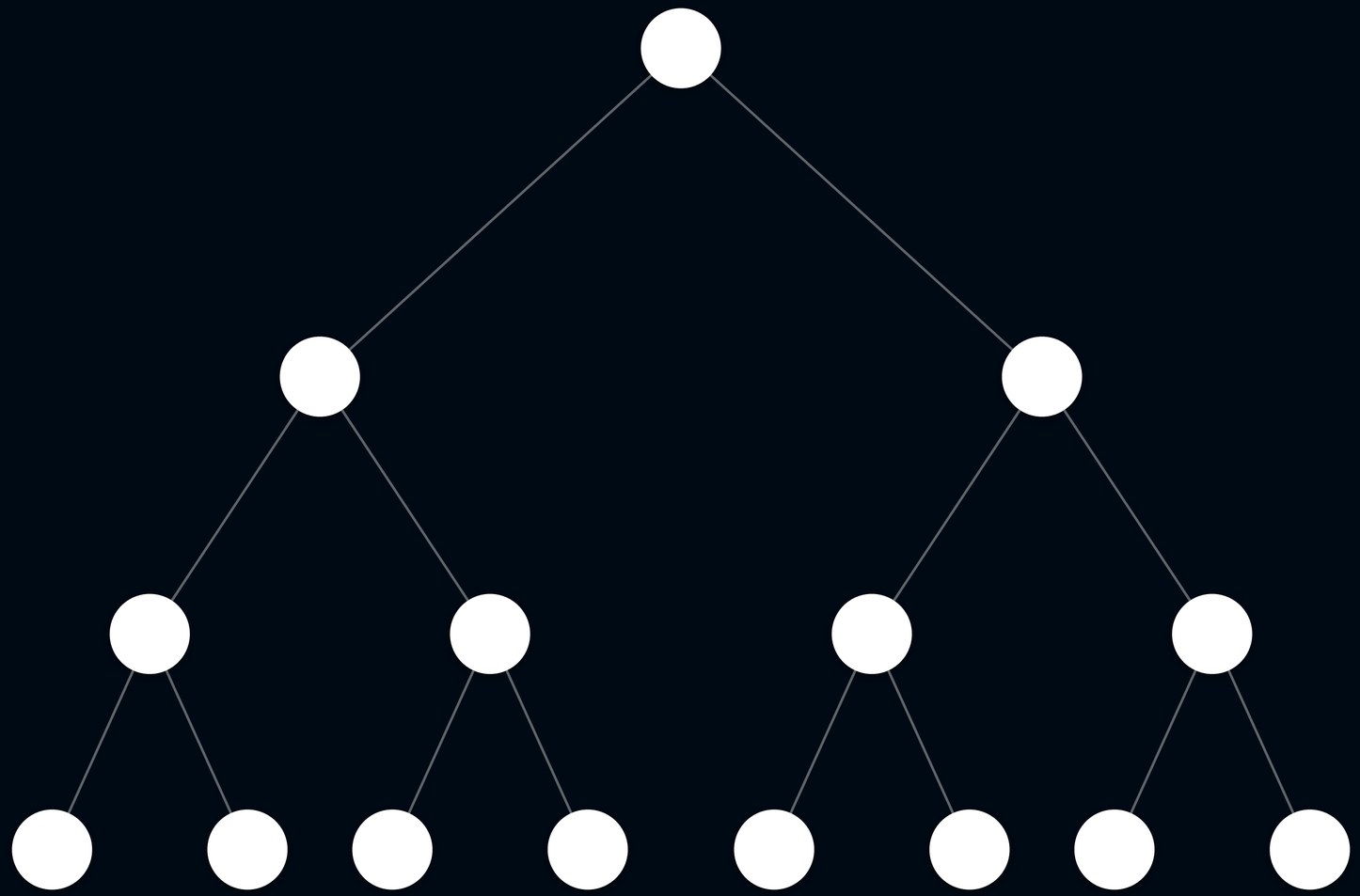
THE MIRROR

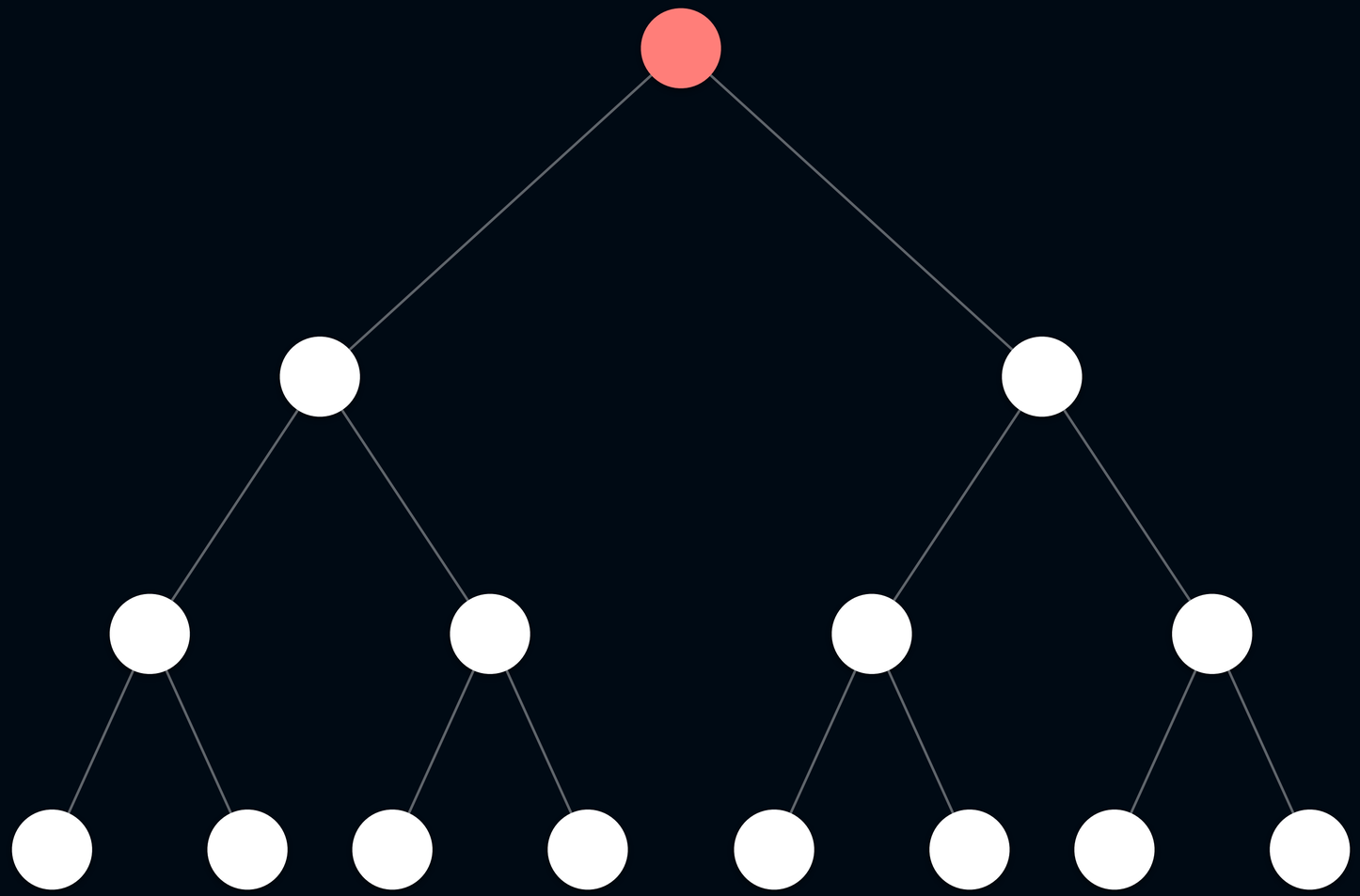
THE MIRROR

Angular now has **built-in** state management with Signals

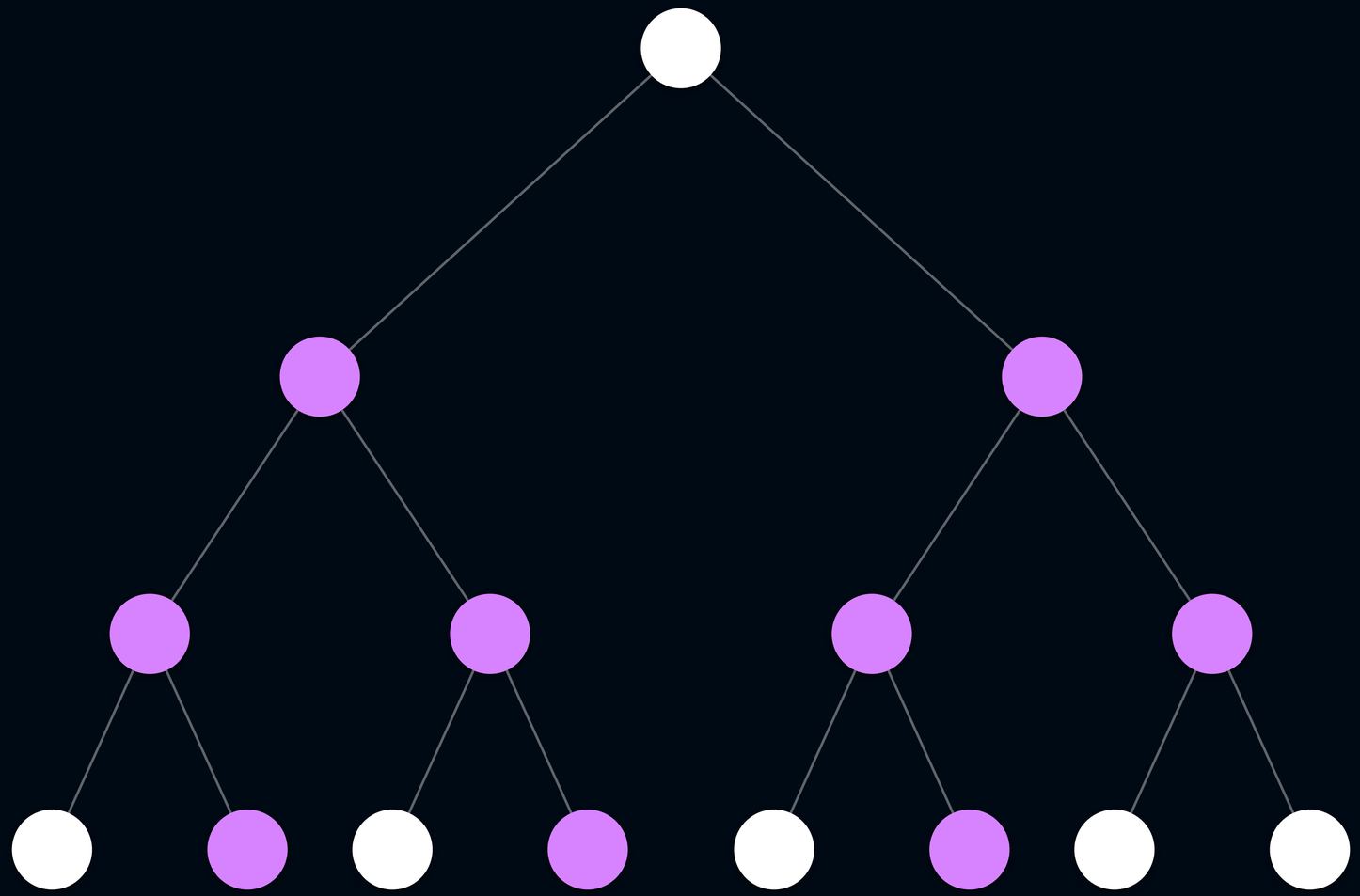
Do we still need NgRx?

Demo

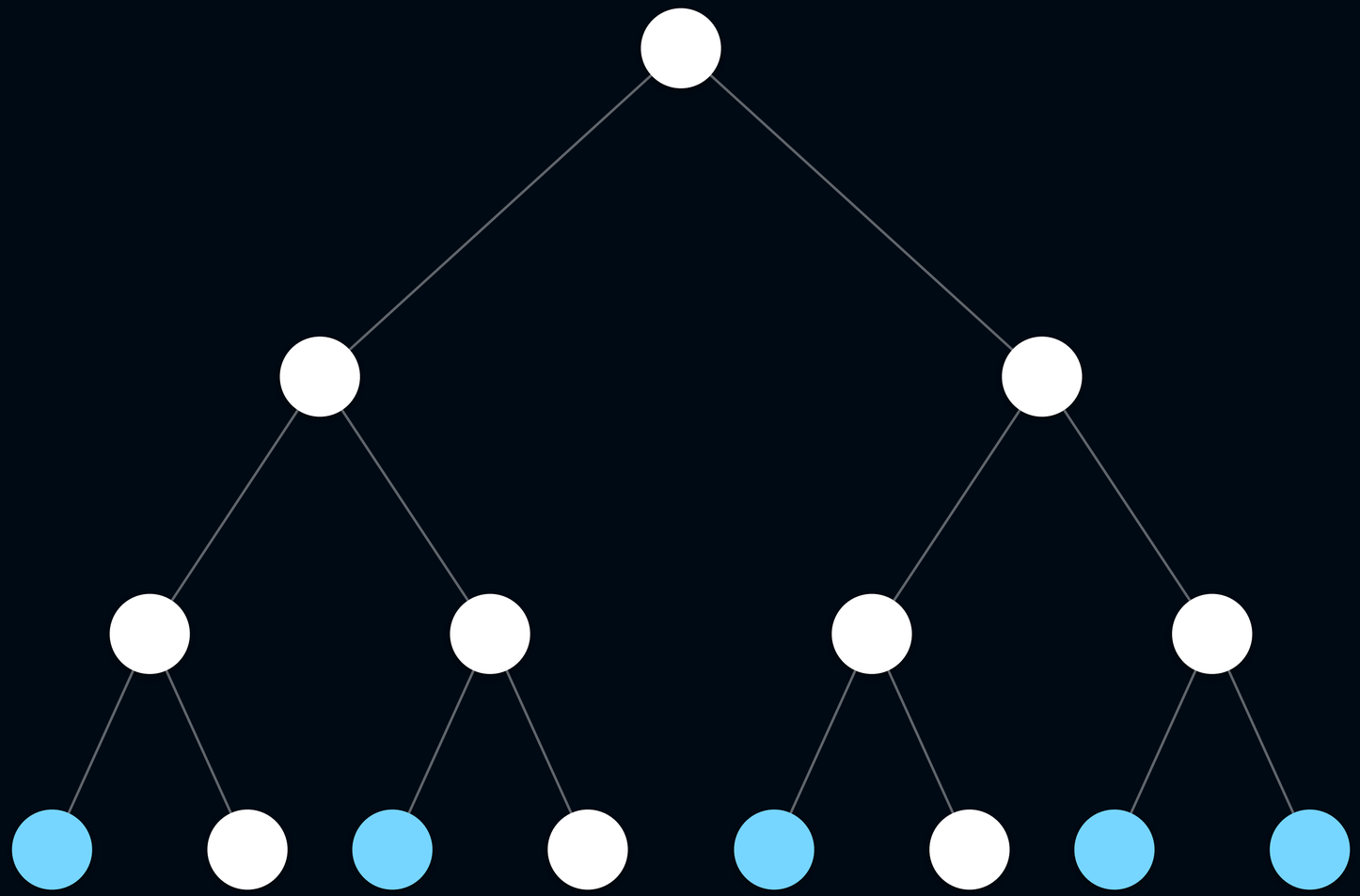




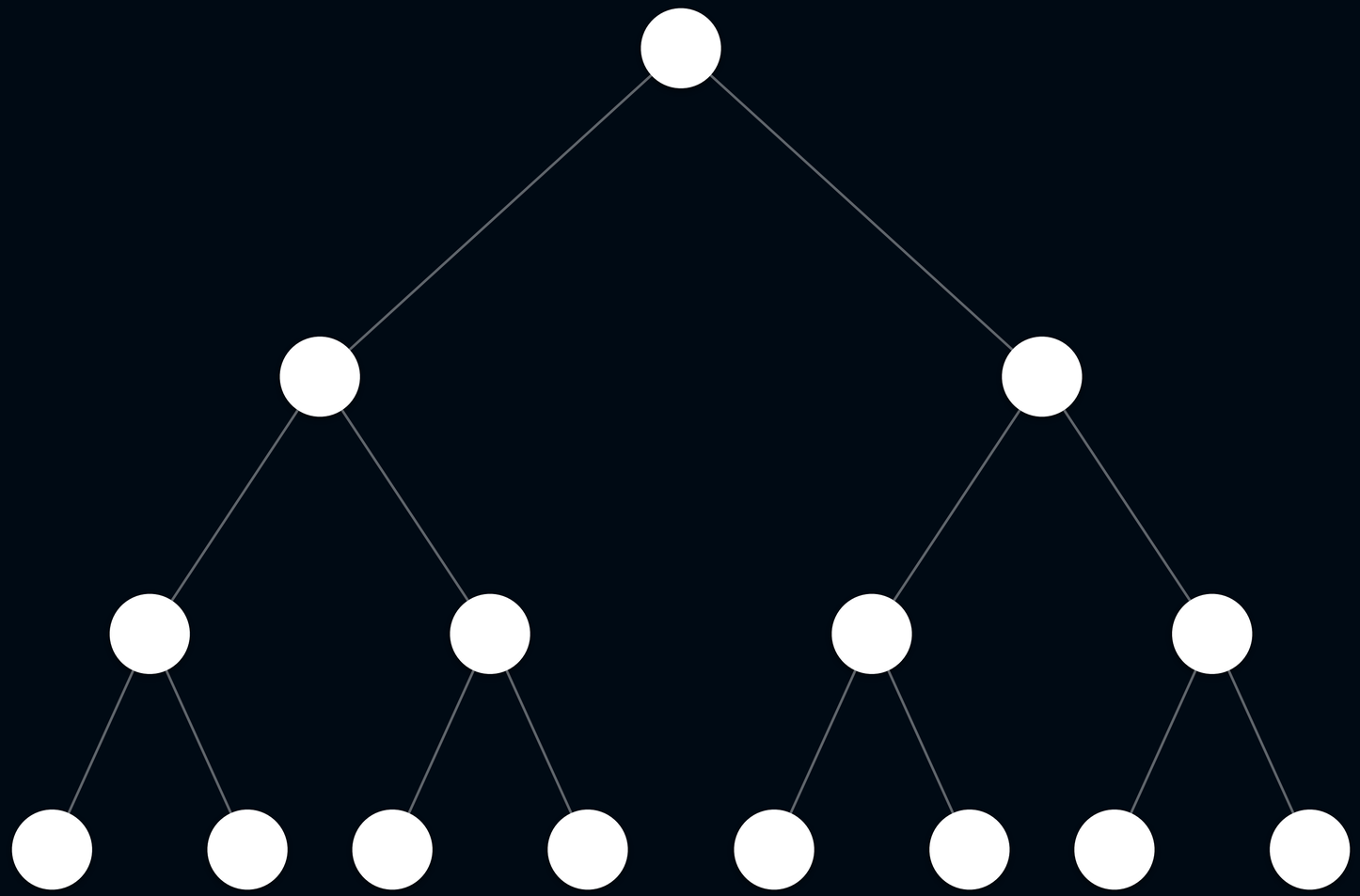
@ngrx/store



@ngrx/signals



signal()



Lift state as needed

THE MIRROR

◇

THE BALANCER

◇
◇

THE CONDUCTOR

◇◇◇

THE MIRROR

◇

THE BALANCER

◇
◇

THE CONDUCTOR

◇◇◇

THE MIRROR

◇
◇
◇

THE FLOW

THE FLOW

Angular now has a **built-in** reactivity primitive

Do we still need RxJS?

Ben Lesh
@BenLesh

Consider the following async/await.

Pretty straight forward, right? Sure.

Here's what I see:

1. What happens when a user clicks this button again before the first async handler is done?
2. What happens if both awaits in the first call succeed, but the second await in the second call fails or hangs?
3. What happens if the second await in the first call is a lot slower than both awaits in the second call?
4. If we add some sort of cancellation to this, how do we manage that? It has to be passed to each promise-returning function, and as promises aren't really cancellable, it must reject in order to interrupt the async function processing....

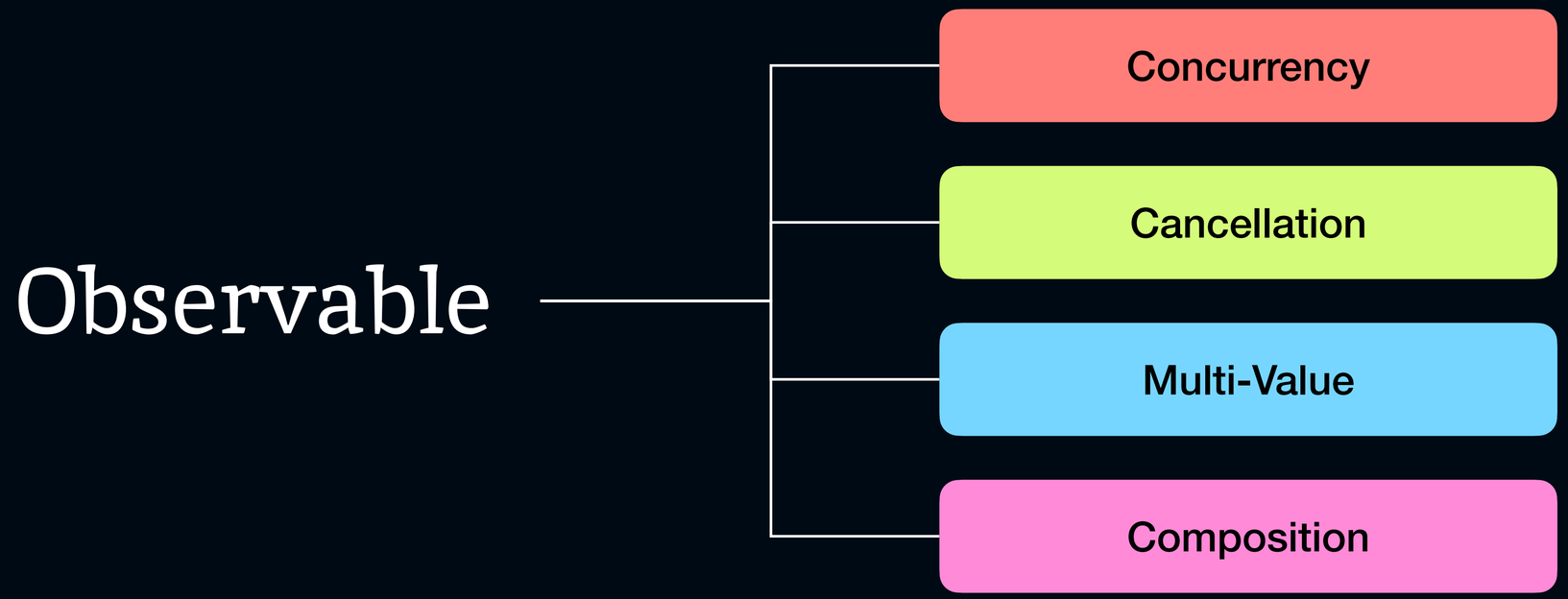
In short, async/await is lipstick on a pig. The pig being asynchronous programming in general.

Async programming is NEVER easy. It's NEVER straightforward.

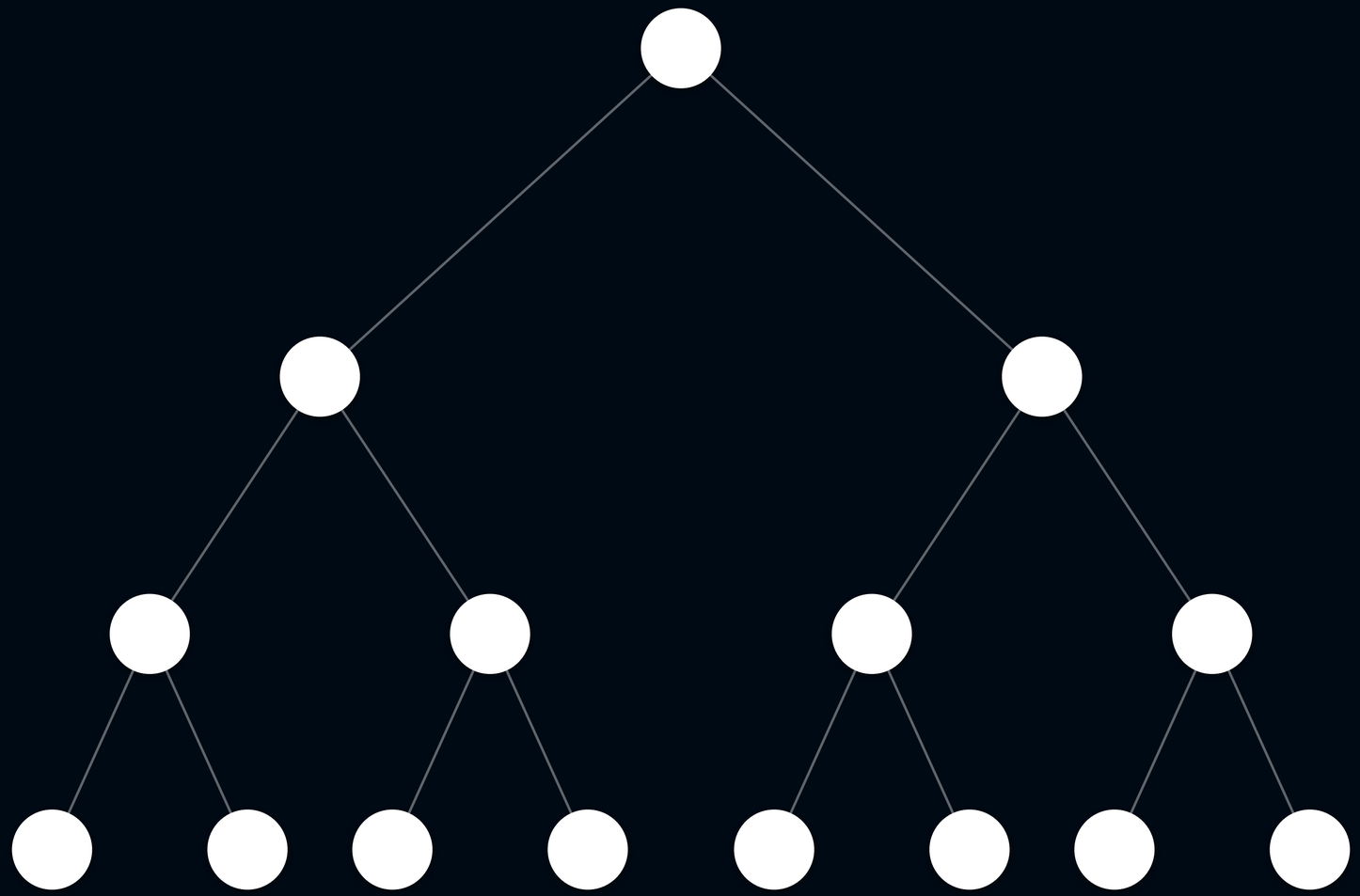
```
<button onClick={async () => {
  const bidsAndPrices = await calculateBidsAndPrices();
  const userWatchLists = await getAssociatedWatchLists(bidsAndPrices);
  setWatchListDisplay(userWatchList, bidsAndPrices);
}}>Click Me Fast</button>
```

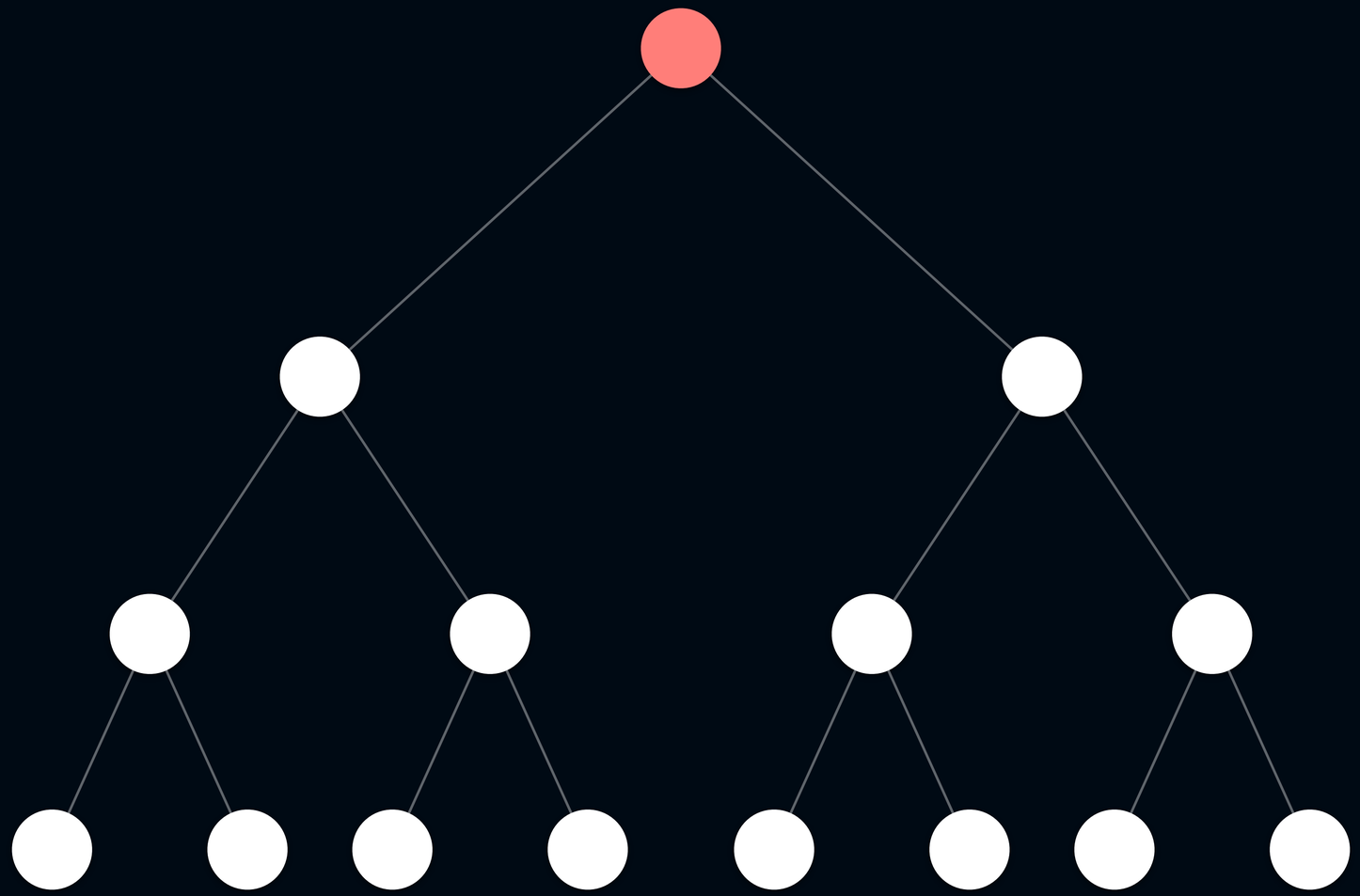
Async programming is **NEVER** easy. It's **NEVER** straightforward.

Demo

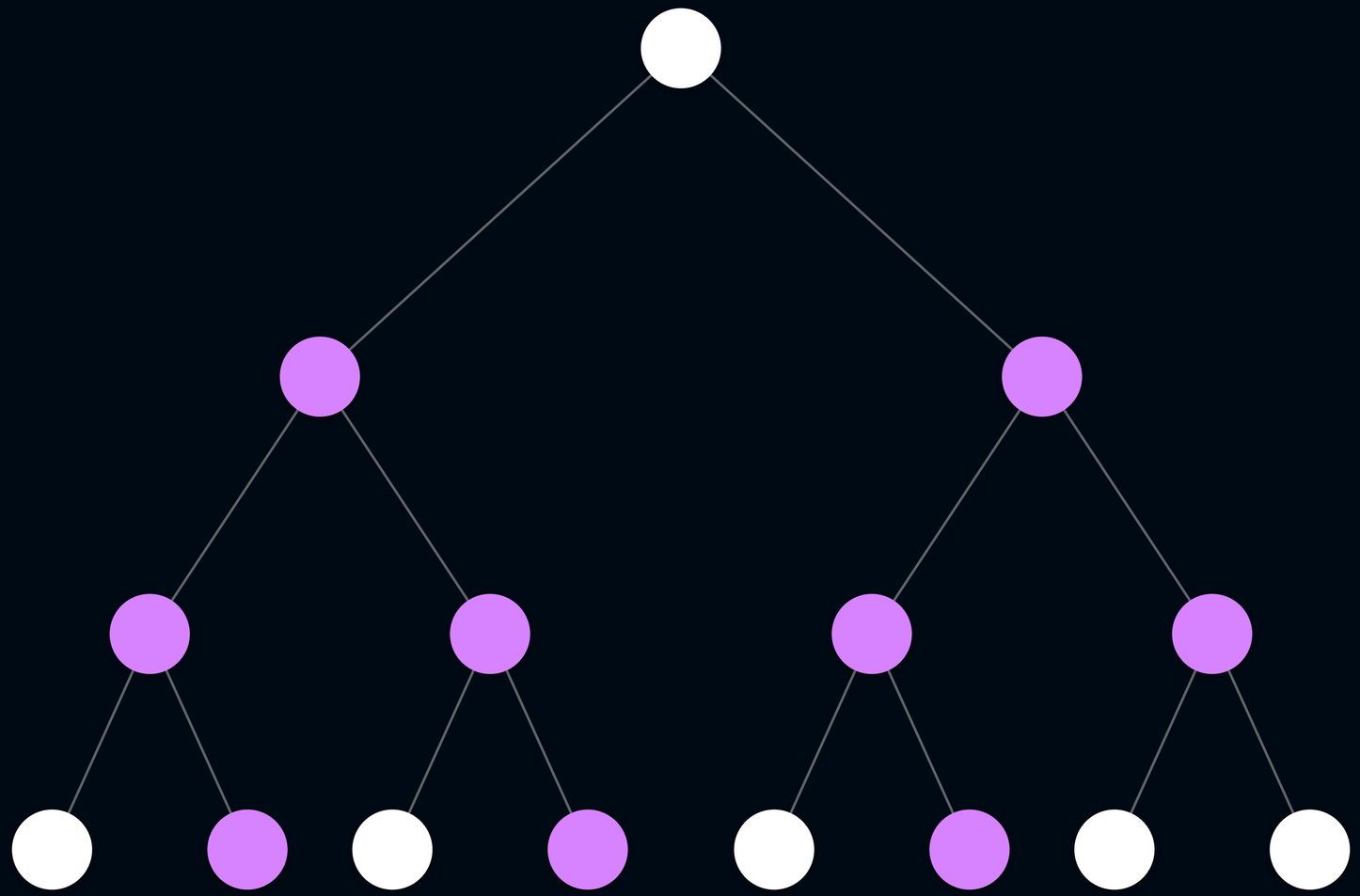

Observable
Concurrency
Cancellation
Multi-Value
Composition

Demo

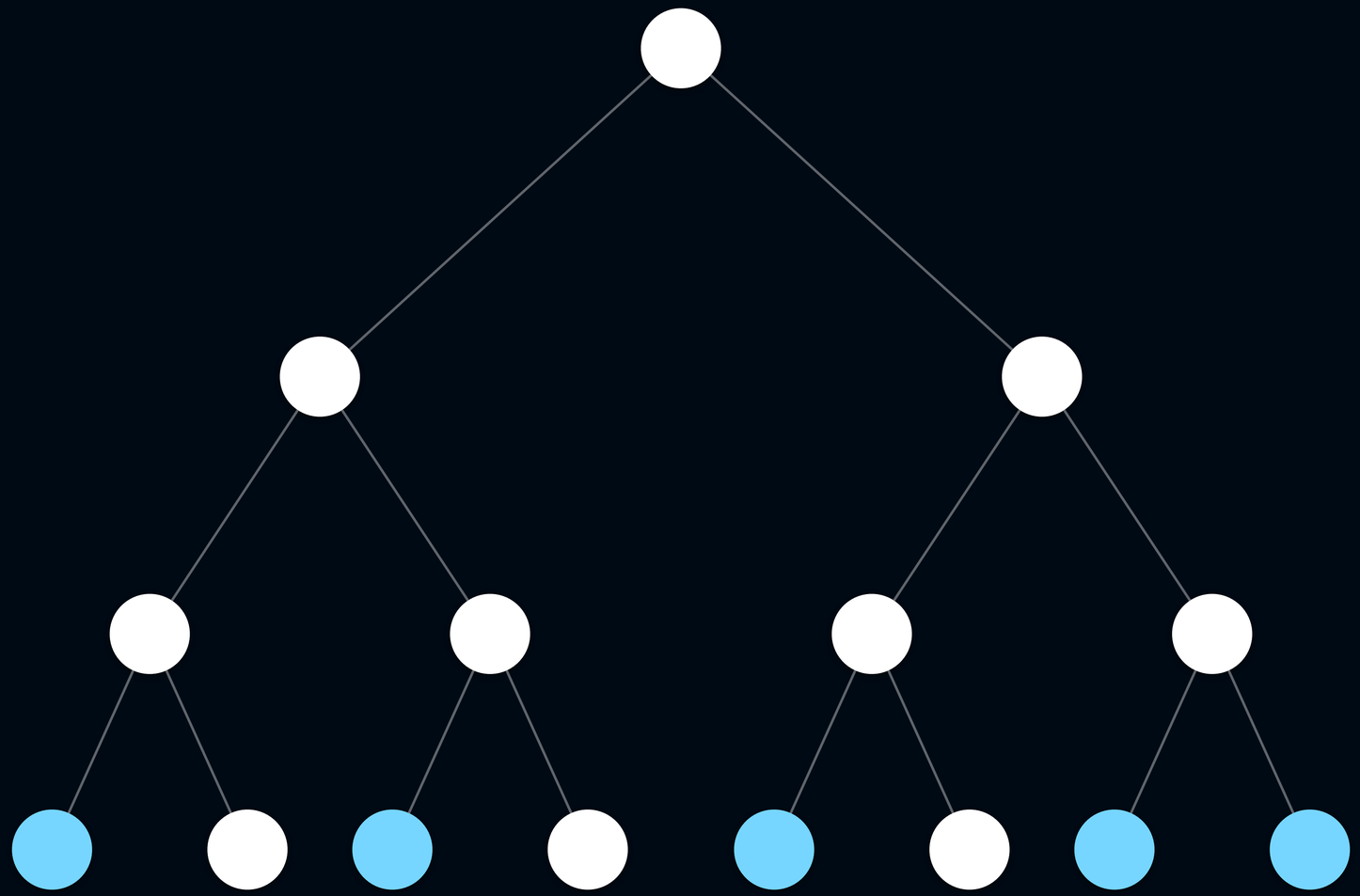




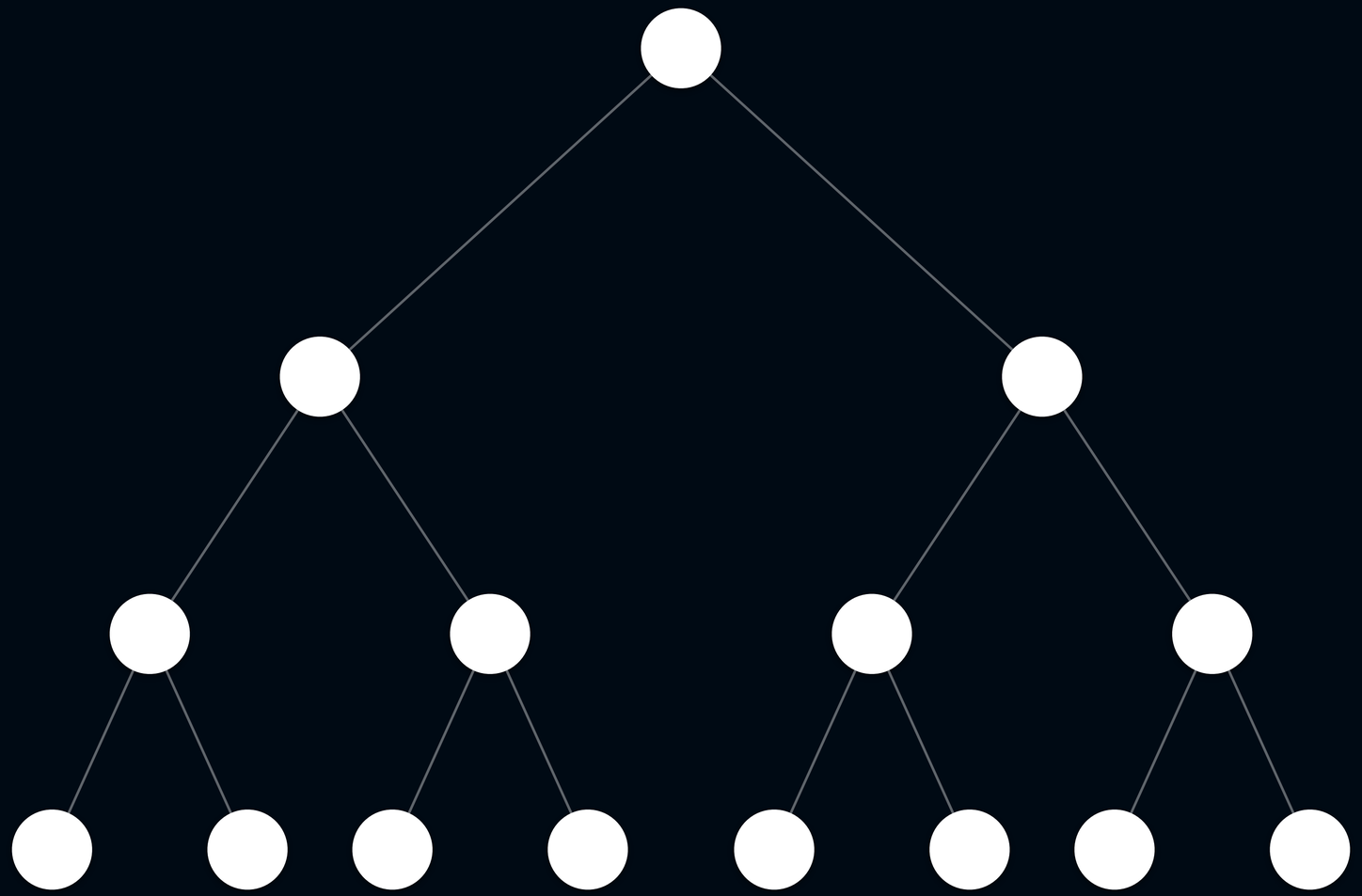
@ngrx/effects



@ngrx/signals



effect()



Lift effects as needed

THE FLOW

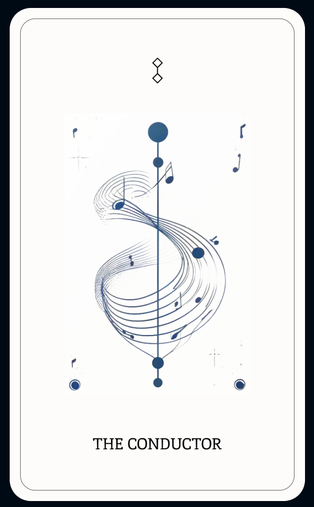
THE BALANCER

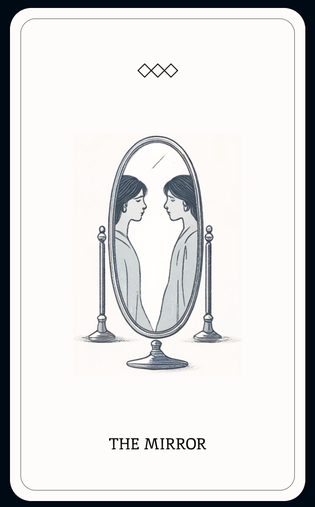
THE CONDUCTOR

THE MIRROR

THE FLOW


THE BALANCER


THE CONDUCTOR


THE MIRROR


THE FLOW

THE BALANCER

THE CONDUCTOR

THE MIRROR

THE FLOW

THE BALANCER

THE CONDUCTOR

THE MIRROR

THE FLOW

THE BALANCER

THE CONDUCTOR

THE MIRROR

THE FLOW

◇

THE BALANCER

◇
◇

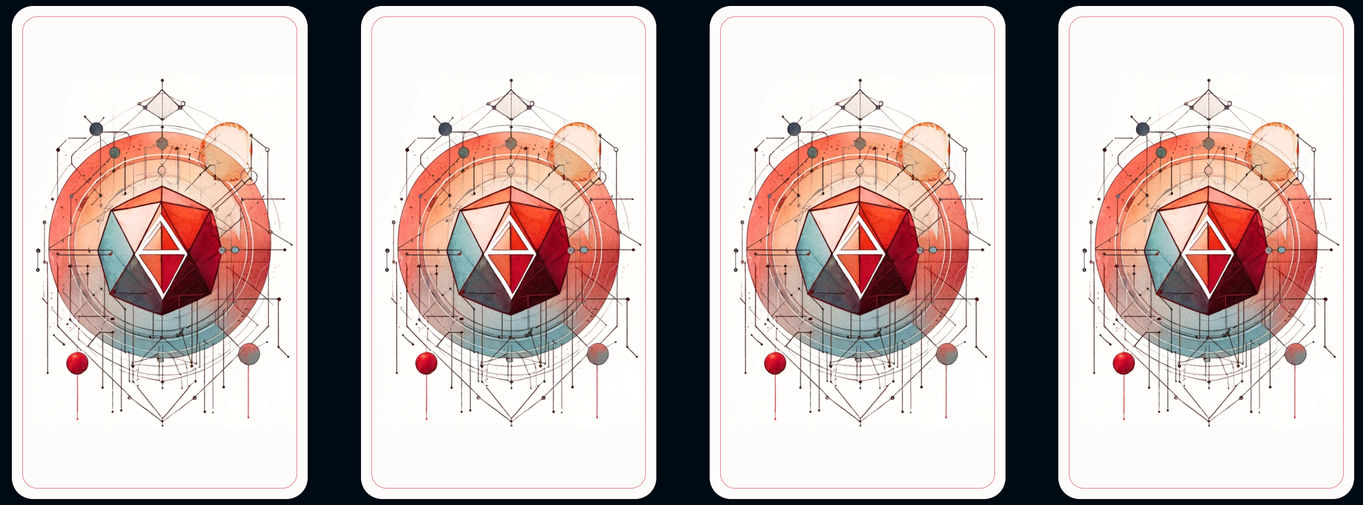
THE CONDUCTOR

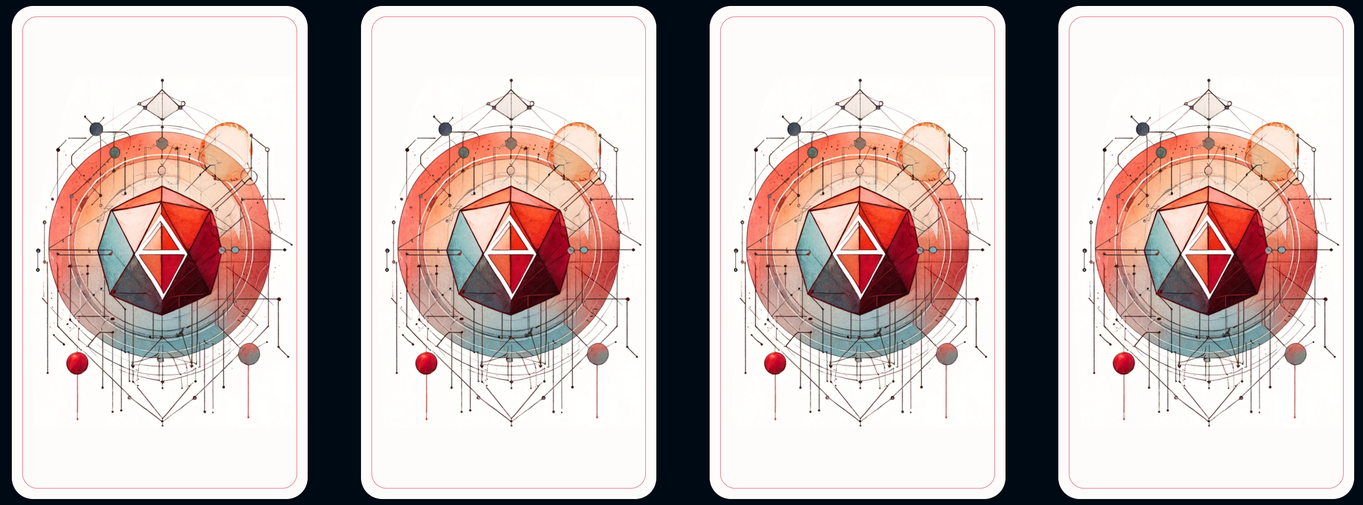
◇◇◇

THE MIRROR

◇
◇
◇

THE FLOW





github.com/mikeryandev/ng-belgrade-2024

Thank you