

GDC

March 20-24, 2023
San Francisco, CA

Fixing Bugs by Cloning Them in *The Last Clockwinder*

Matt Blair, Pontoco

#GDC23

Hi! I'm Matt Blair.

Founder at Pontoco

- Game design
- Engineering
- Management





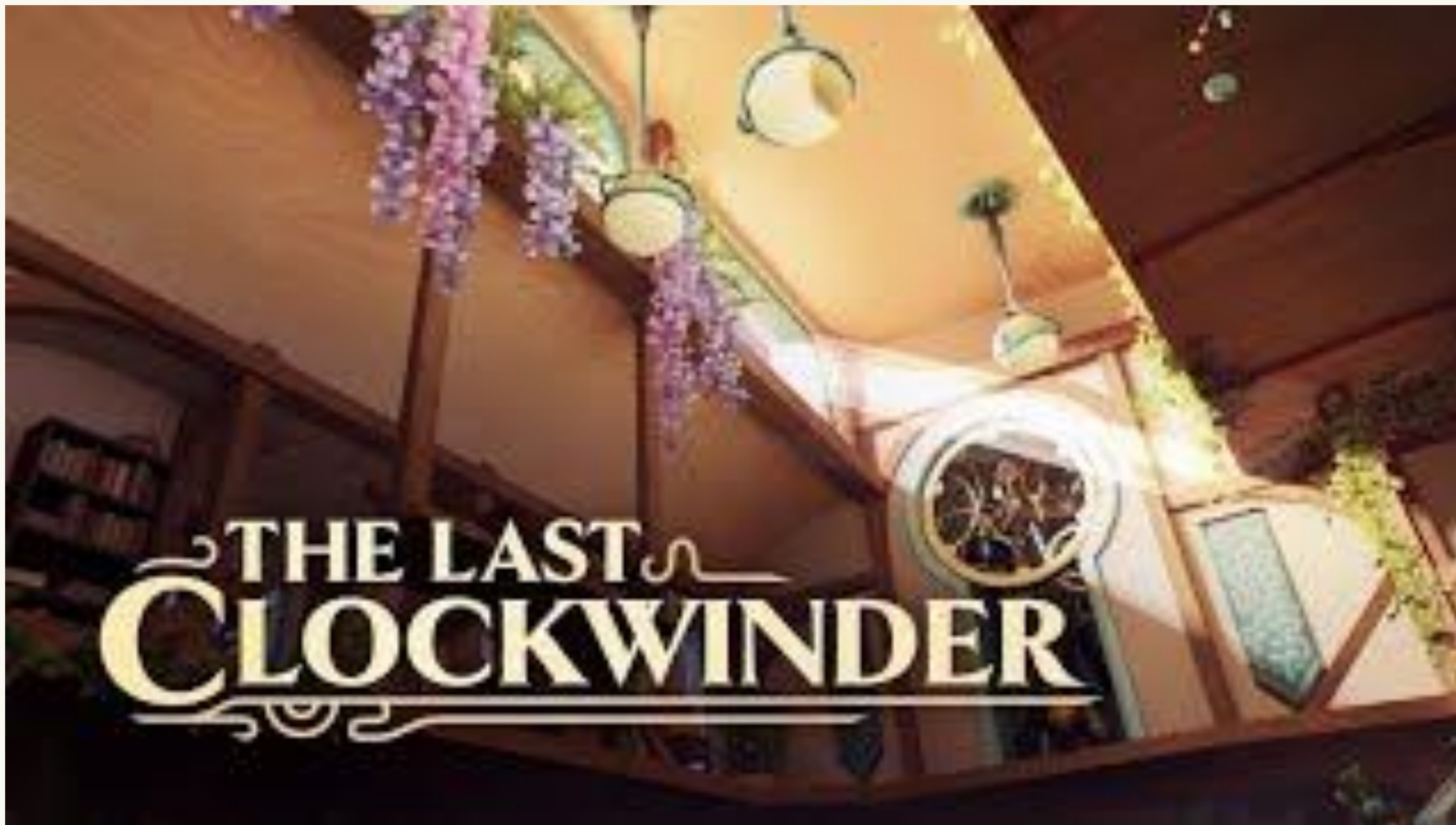
THE LAST CLOCKWINDER



WINNER
∞ Meta Best of Quest
BEST GAME
of 2022

WINNER
ROAD TO VR
BEST PC
VR GAME
2022

WINNER
HC **HARDCORE**
GAMER
BEST
VR GAME
2022



How did we pull this off with just two (part time) engineers?

Input Replay system

- All bugs reproducible
- Fast iteration time in VR
- Automated gameplay testing
- Debug hardware issues without the hardware

Part 1: Fulfilling the gameplay promise

Part 2: Turning the mechanic into the workflow

Part 1:

Fulfilling the gameplay promise

The Promise of *Clockwinder*

- Puzzle-automation game with looping clones
- Mechanic should be as generous as possible
- Clones reproduce the player's actions perfectly, every time

“If I can do something, then my clone can do it too” *

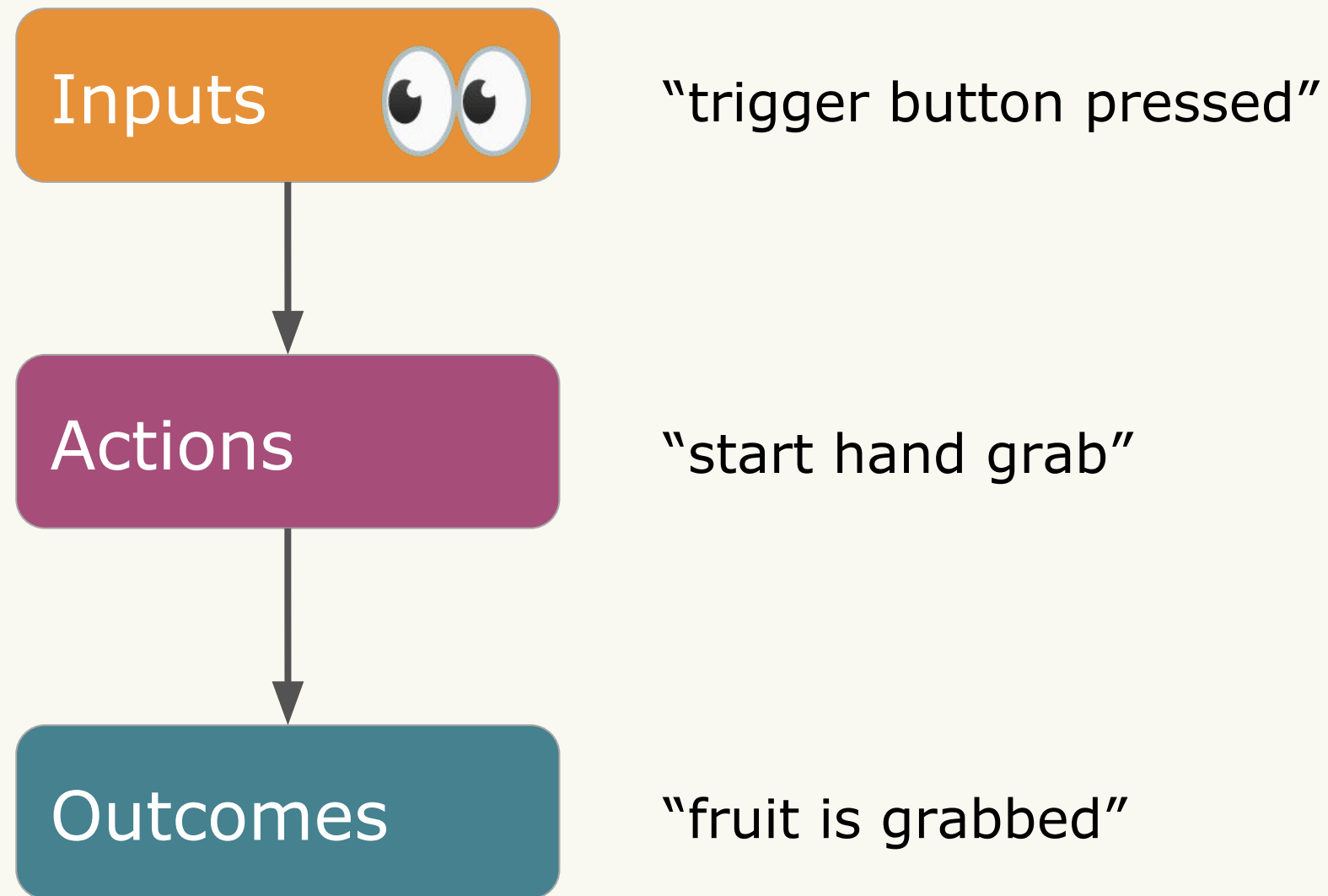
* = Some terms and conditions may apply

Cloning Strategy

Two paths:

1. Record player *actions* and apply them like an animation
 - Only process necessary data
-  2. Record player *inputs* and simulate the resulting actions
 - No need to define which data is necessary

Cloning Strategy



Cloning: Inputs

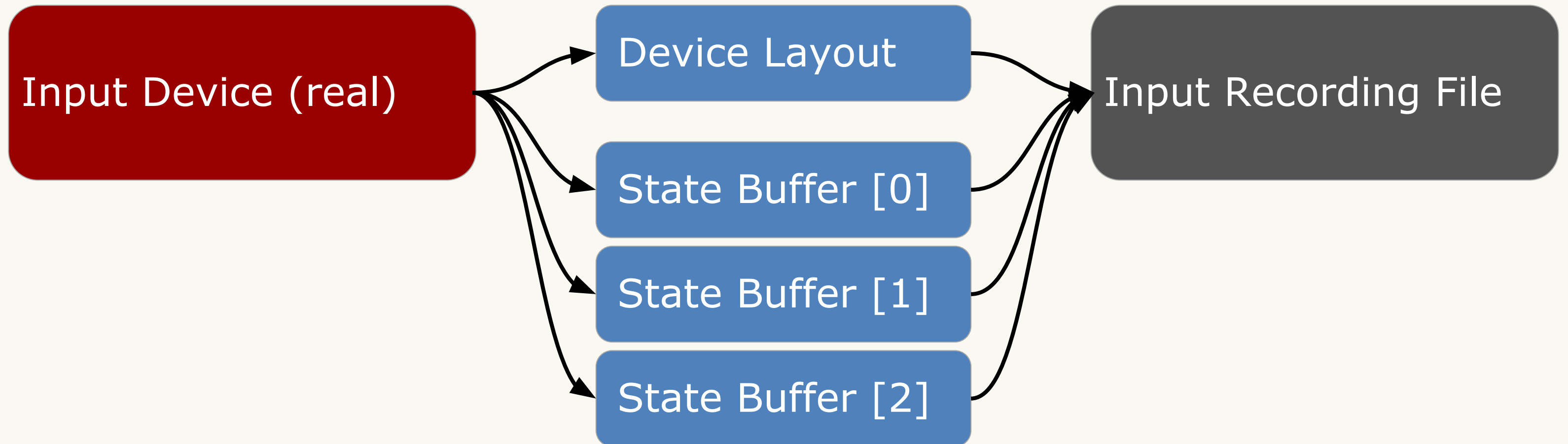
Unity InputSystem package

<https://github.com/Pontoco/InputSystem/tree/asg/v1.4.2>

We ❤️ InputSystem

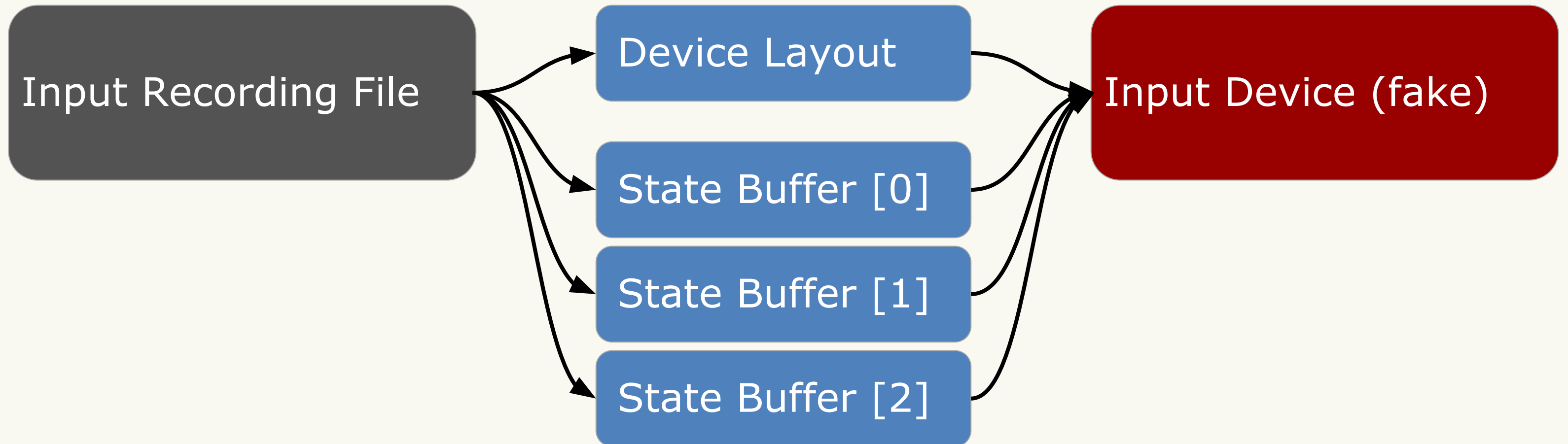
Cloning: Inputs

Recording inputs

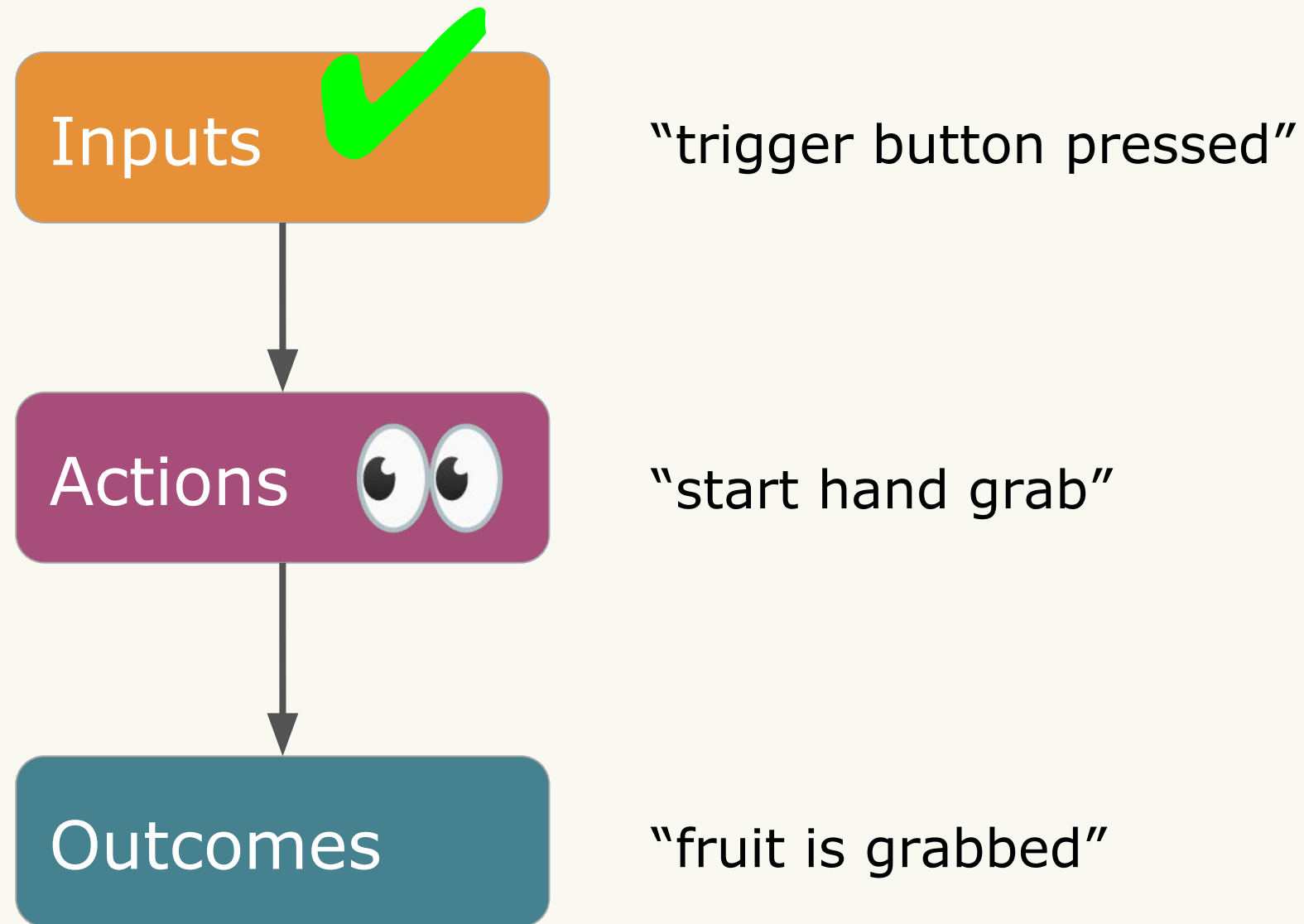


Cloning: Inputs

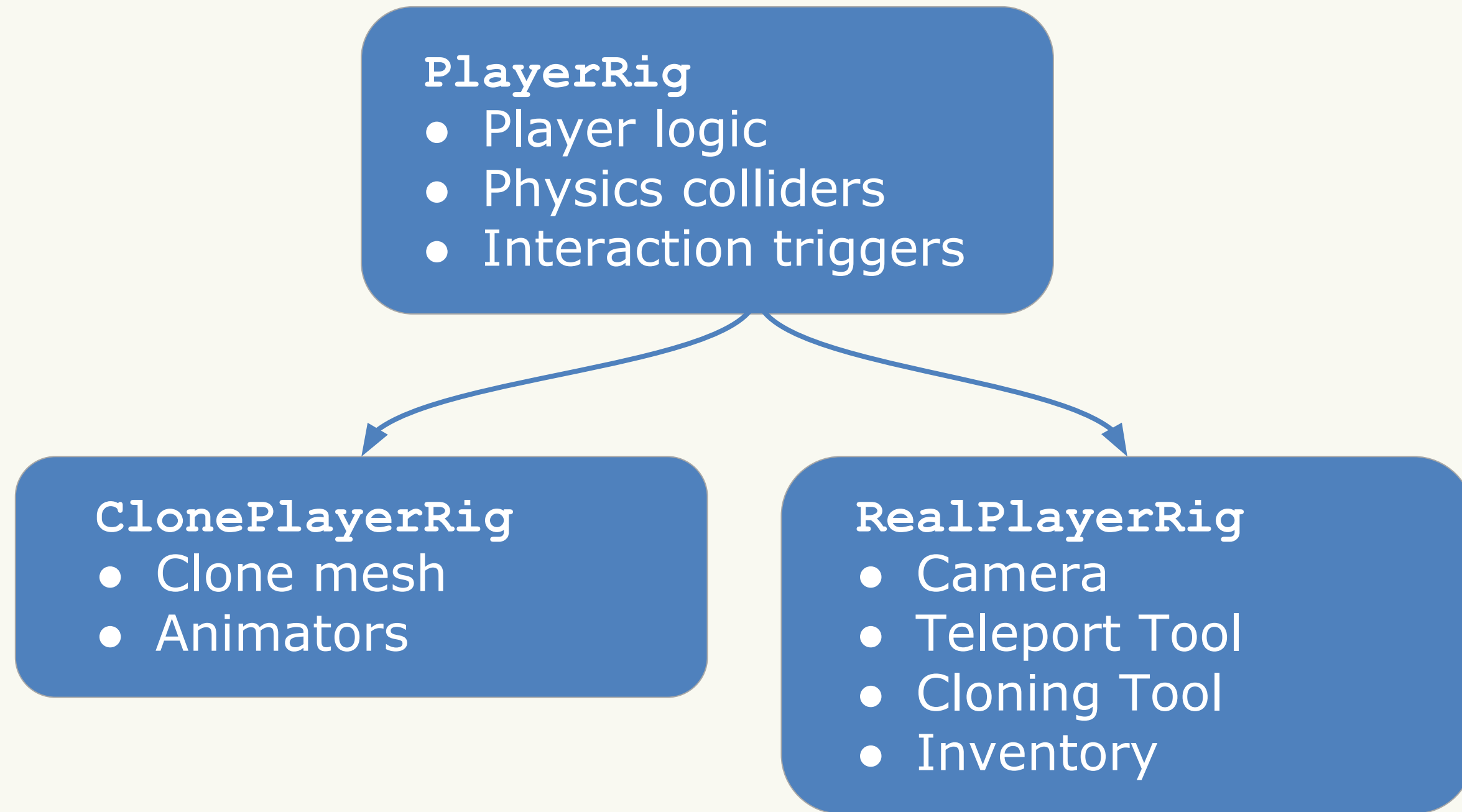
Replaying inputs



Cloning Strategy



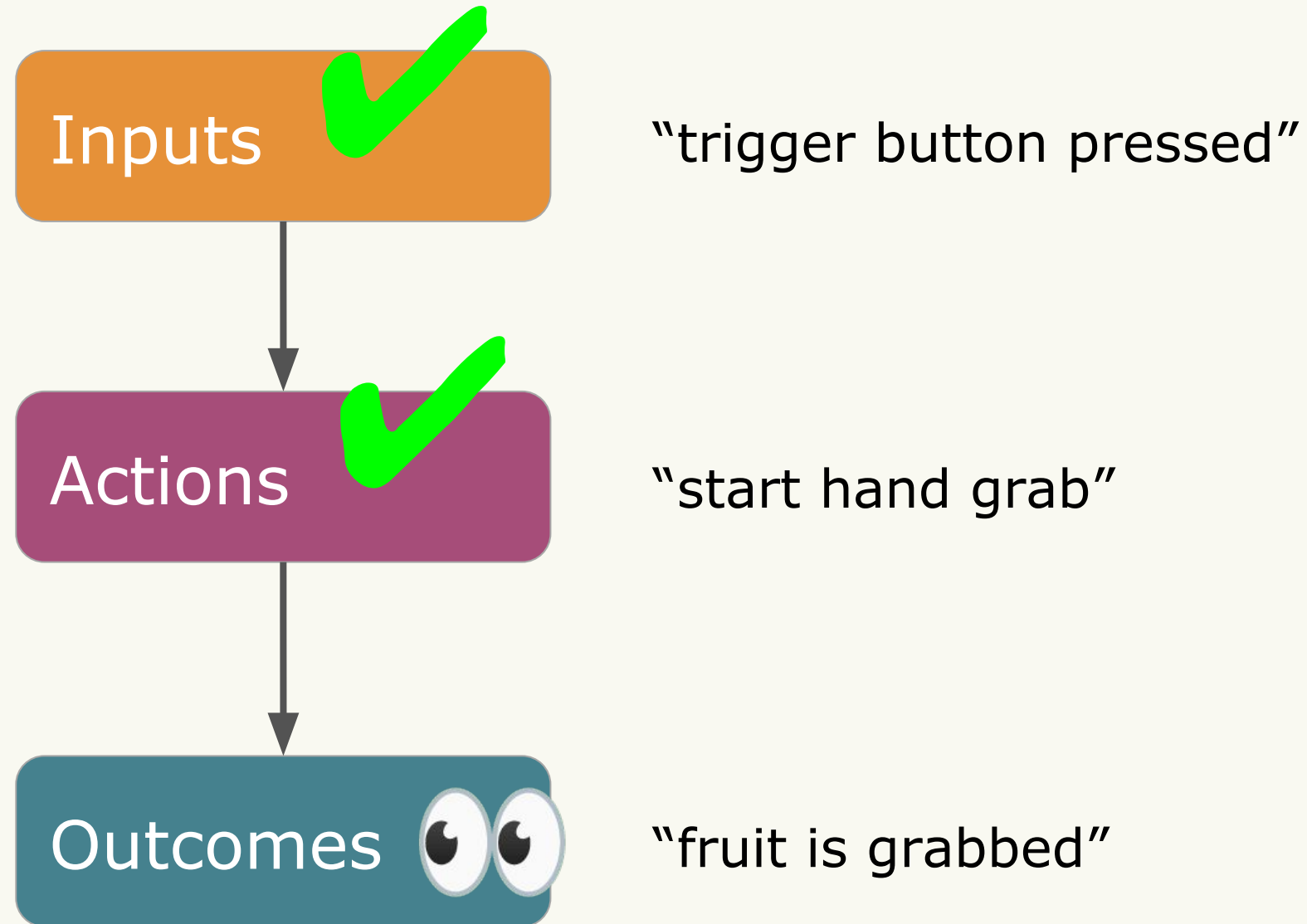
Cloning: Actions



Cloning: Actions

When the player is able to do something, clones are able to do it by default.

Cloning Strategy

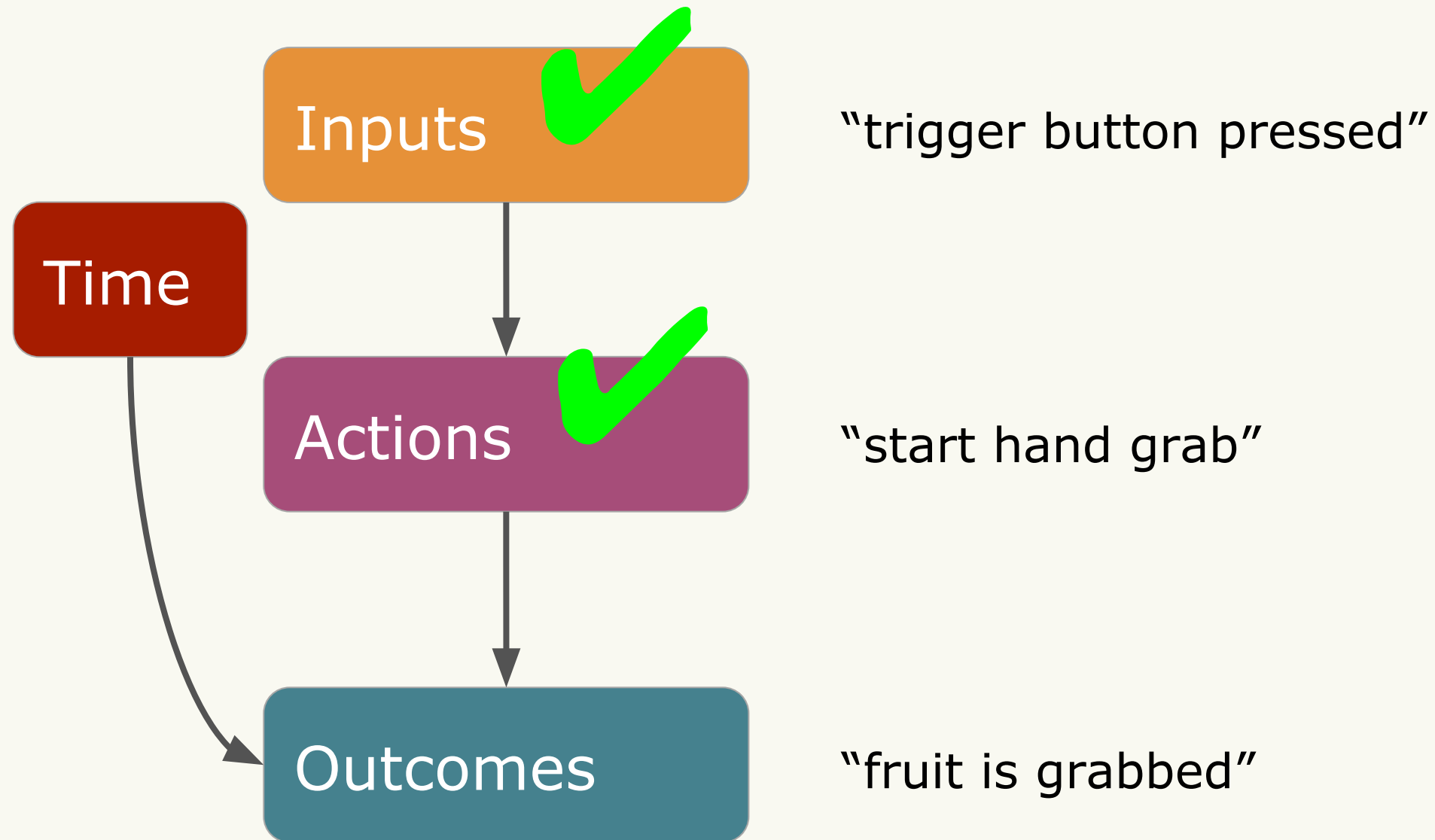


Cloning: Outcomes

What makes clones not do the same thing as the player?

Anything that is not part of the recorded inputs.

Cloning Strategy



Cloning: Outcomes



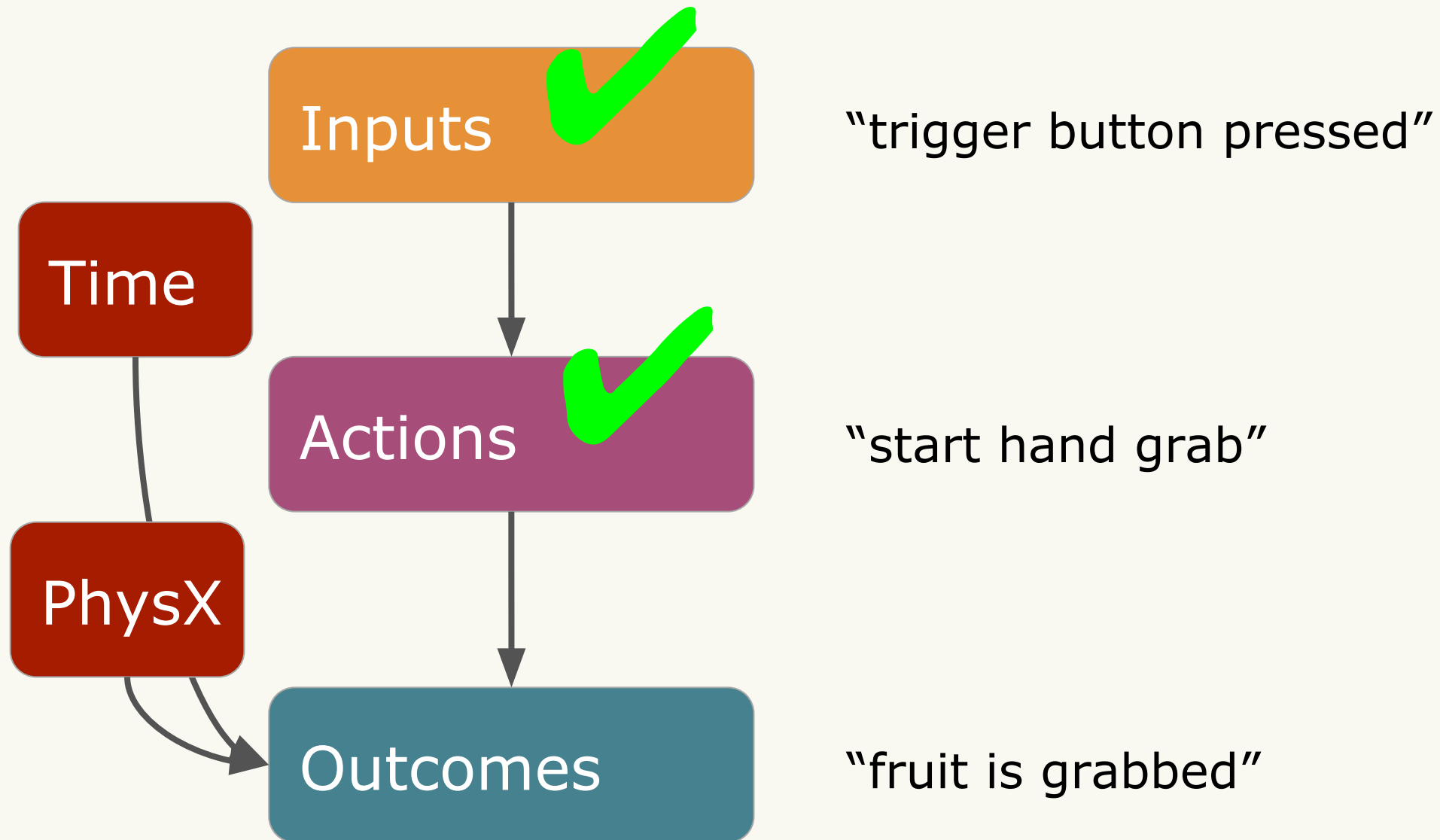
Cloning: Outcomes

Can we record frame times?

No, because we have to update many clones in the same frame.

- All clones recorded and replayed at the same, fixed frame rate.
- All game logic goes in `FixedUpdate`. 🦀

Cloning Strategy

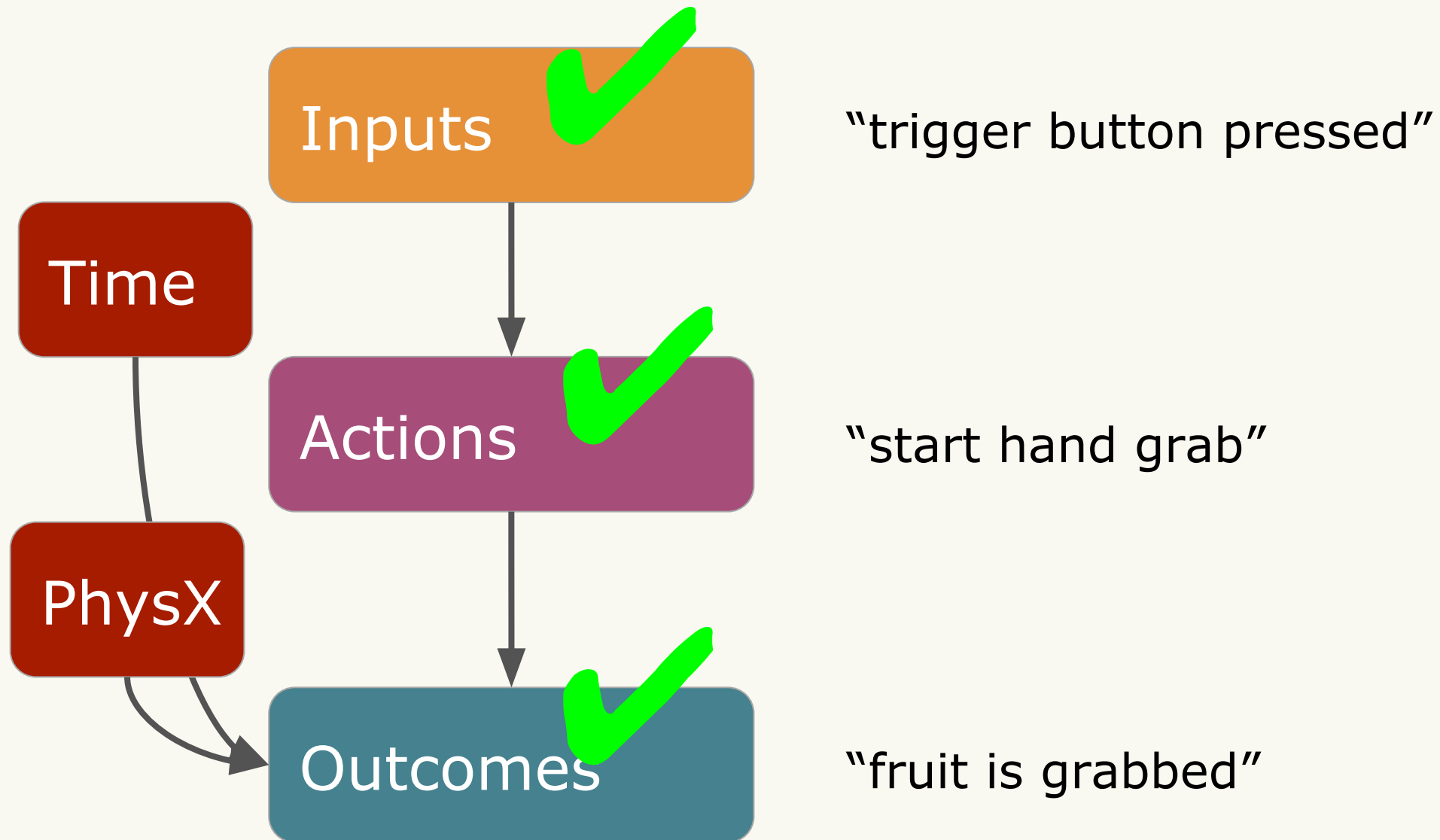


Cloning: Outcomes

Outcome of certain collisions depends on physics engine state.

- For important collisions, skip physics and use a custom solver. 🦀

Cloning Strategy





The Promise of *Clockwinder*

- Puzzle-automation game with looping clones
- Mechanic should be as generous as possible
- Clones reproduce the player's actions perfectly, every time

“If I can do something, then my clone can do it too” *



* = Some terms and conditions may apply

Part 2:

The mechanic becomes the workflow



Session Replay

We can extend replays to the entire development workflow.

Session Replay

Everything just worked!

... just kidding.

Session Replay: Challenges

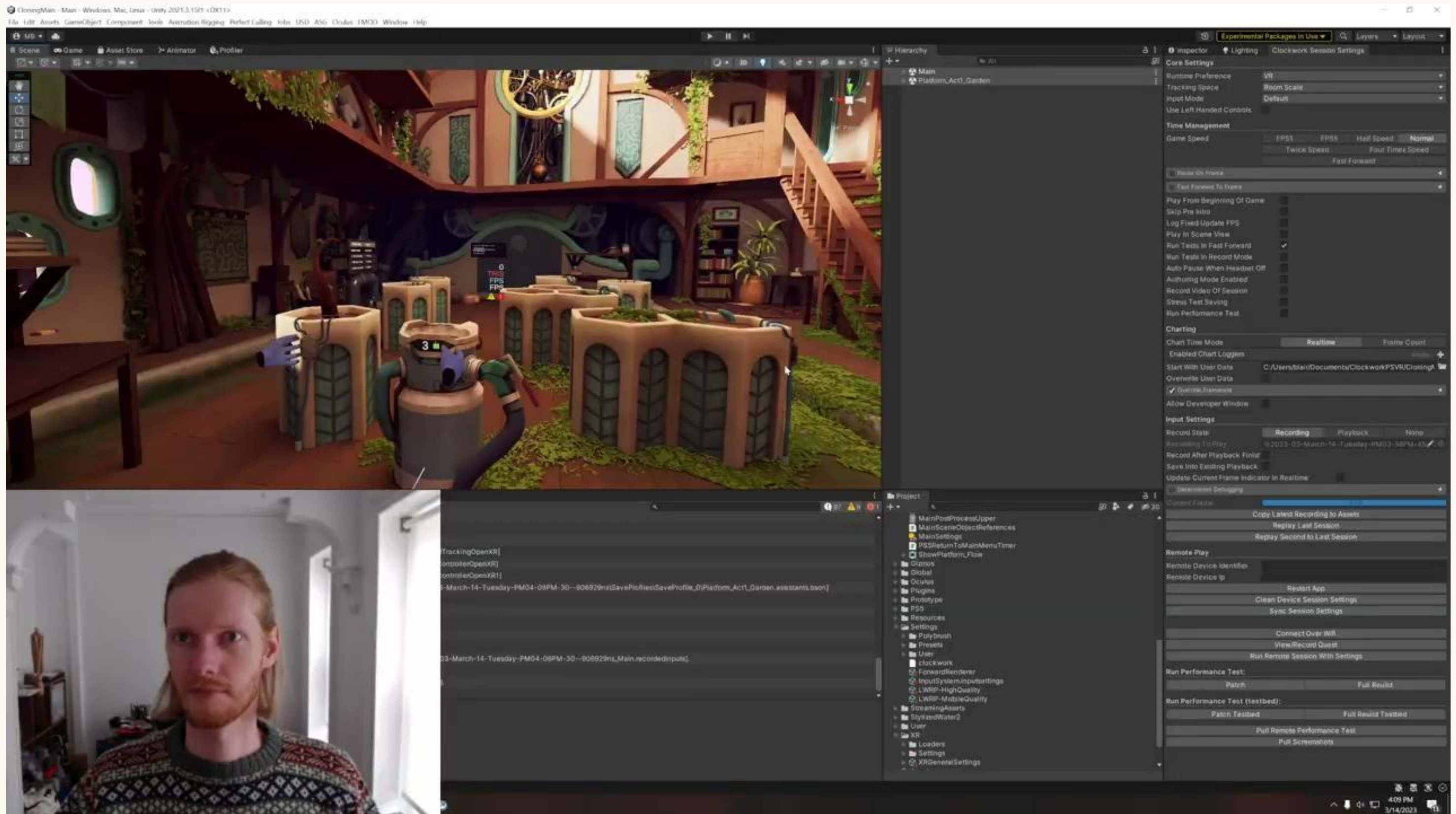
- **Replays need to start with the exact same saved data.**
 - Add tools for copying and overriding save data.

Session Replay: Challenges

- **Replaying a full game session takes a long time.**
 - Add fast-forward modes to session replay. 🦀

Session Replay: Challenges

- **Loading scenes takes variable time.**
 - Record number of loading frames, then simulate same number of frames during replay.



Session Replay: Perks

- **No more phantom bugs!**
 - If we have replay data, we can see the bug again.
 - We can replay the session again to verify the fix.

Session Replay: Perks

- **Fast iteration on features!**
 - Record a session where you play the part you're working on.
 - Replay that session over and over as you iterate.

Session Replay: Perks

- **Recordings become automated tests!**
 - Record a player doing a sequence of actions.
 - Replay the recording in the test runner to verify the outcome.

Session Replay: Perks

- **Simulate devices that we don't actually have!**
 - Get a remote player to capture a session recording on their hardware.
 - Replay that session on our own machines and simulate any problems that are occurring with the different hardware.

Next Steps

There are some 🦀 to deal with.

Next Steps



Sessions must be replayed from the beginning.

- Jump to points in the recording using state snapshots (DOTS)

Next Steps

 **Outcome of collisions depends on physics engine state.**

- Use UnityPhysics package

Next Steps

- 🦀 **All game logic must occur in `FixedUpdate`.**
 - Record time with input frames, apply recorded time during replay.

(If you know how to make Unity do this, let's talk!)

Next Steps

Use this in more games!
(i.e. your games)

Art of 'The Last Clockwinder': Visual Cohesion

Anita Tung, Wednesday@10:30am

"Stuck in a Loop": Designing the Clones of 'The Last Clockwinder'

John Austin, Thursday@5:30pm

Thank you!

[@matteblair](https://twitter.com/matteblair)

[@matteblair@xoxo.zone](mailto:matteblair@xoxo.zone)

Interested in what we're doing next?

<https://pontoco.com/jobs>

