Marlowe 101 - Hands On

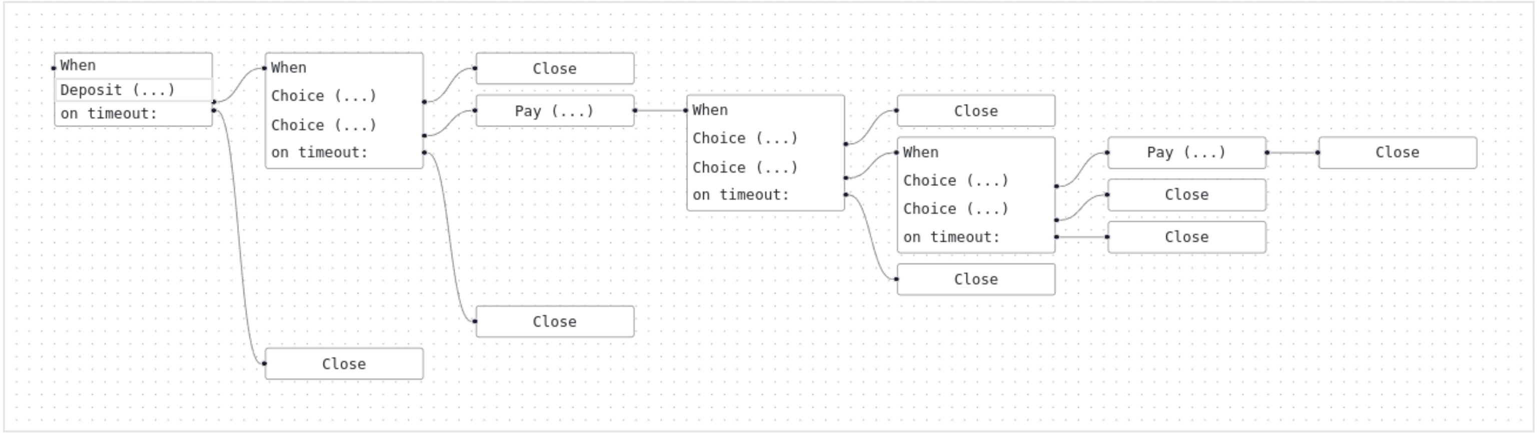
Plan

- How Marlowe Operates
- The Developer Journey
- Building DApps with Marlowe

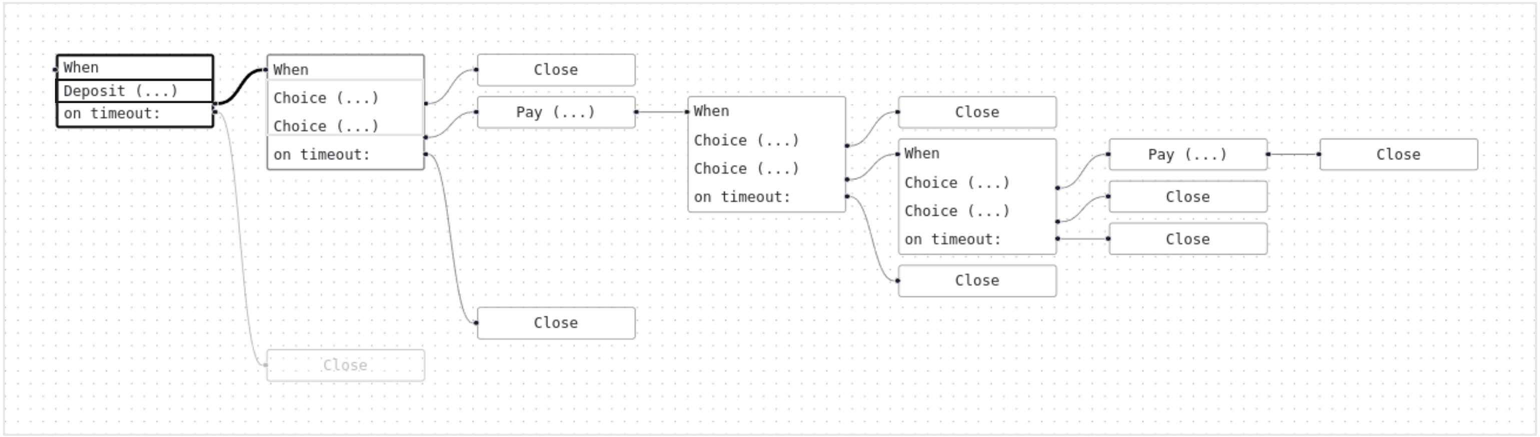
Plan

- **How Marlowe Operates**
- The Developer Journey
- Building DApps with Marlowe

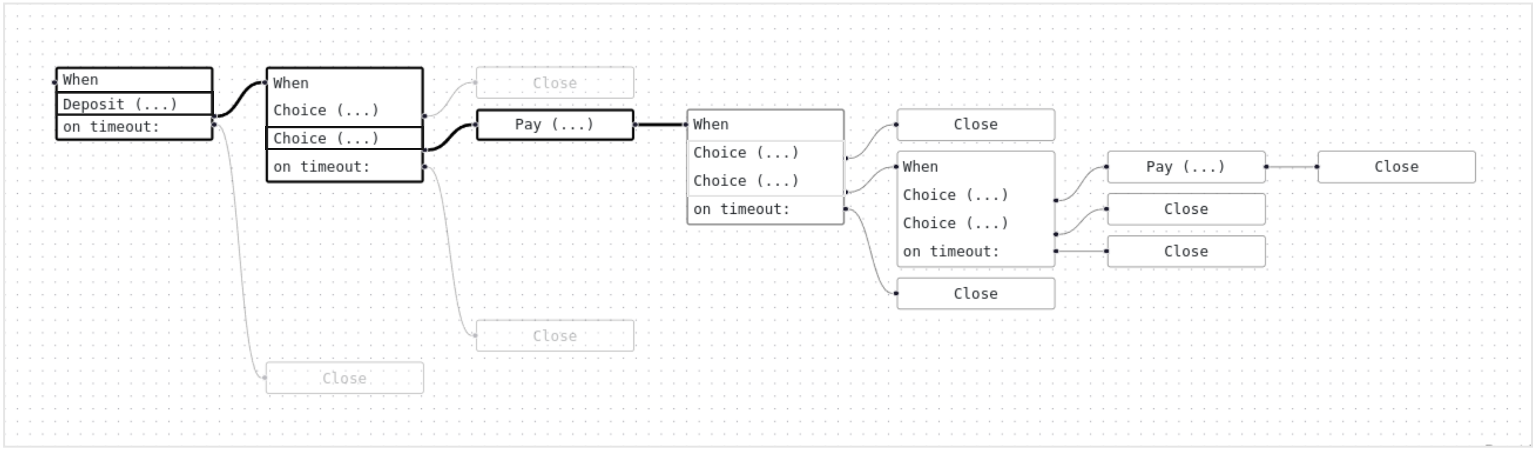
How Marlowe Operates



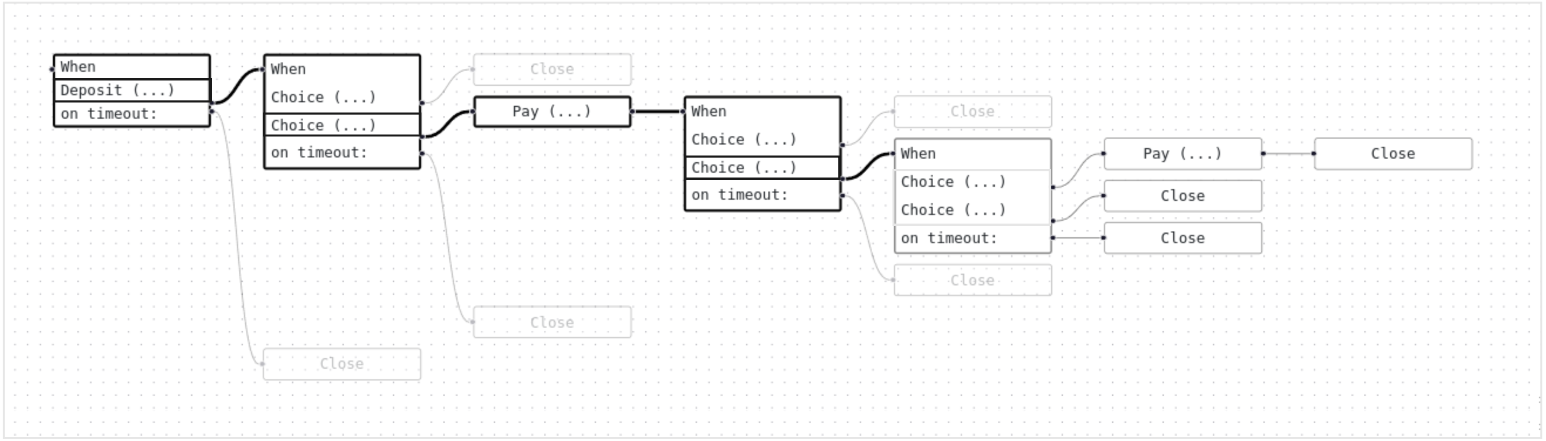
How Marlowe Operates



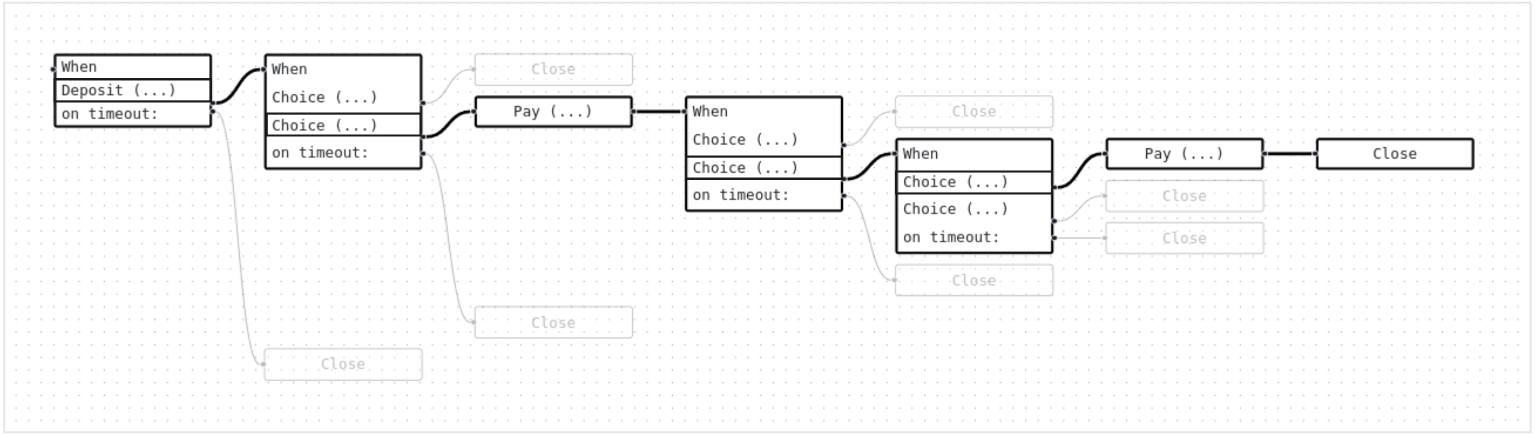
How Marlowe Operates



How Marlowe Operates



How Marlowe Operates



In Marlowe "**When**" construct is our "stopping point".

Plan

- **How Marlowe Operates**
- The Developer Journey
- Building DApps with Marlowe

Plan

- How Marlowe Operates
- **The Developer Journey**
- Building DApps with Marlowe

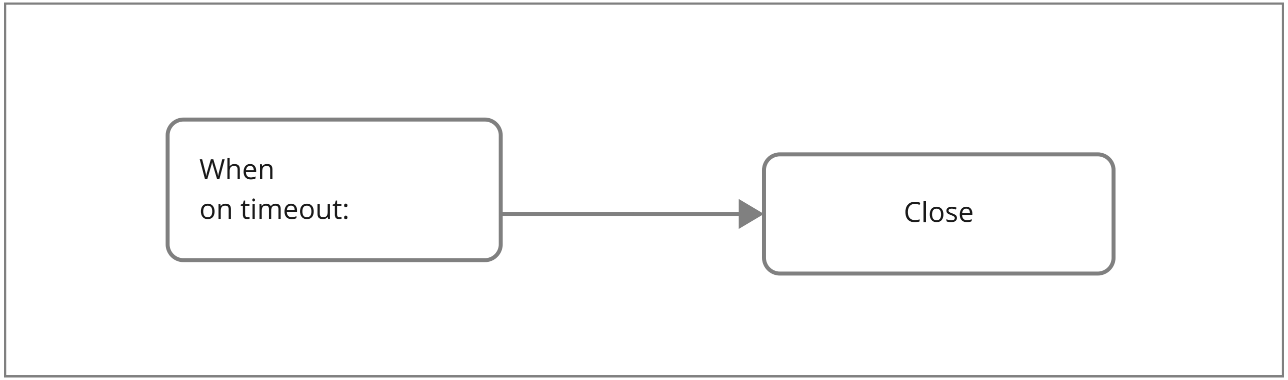
The Developer Journey

The Marlowe Playground: play.marlowe.iohk.io

Exercise: "Await Timeout"

Create a contract which can be closed after a 3 min. timeout.

Exercise: "Await Timeout"



play.marlowe.iohk.io

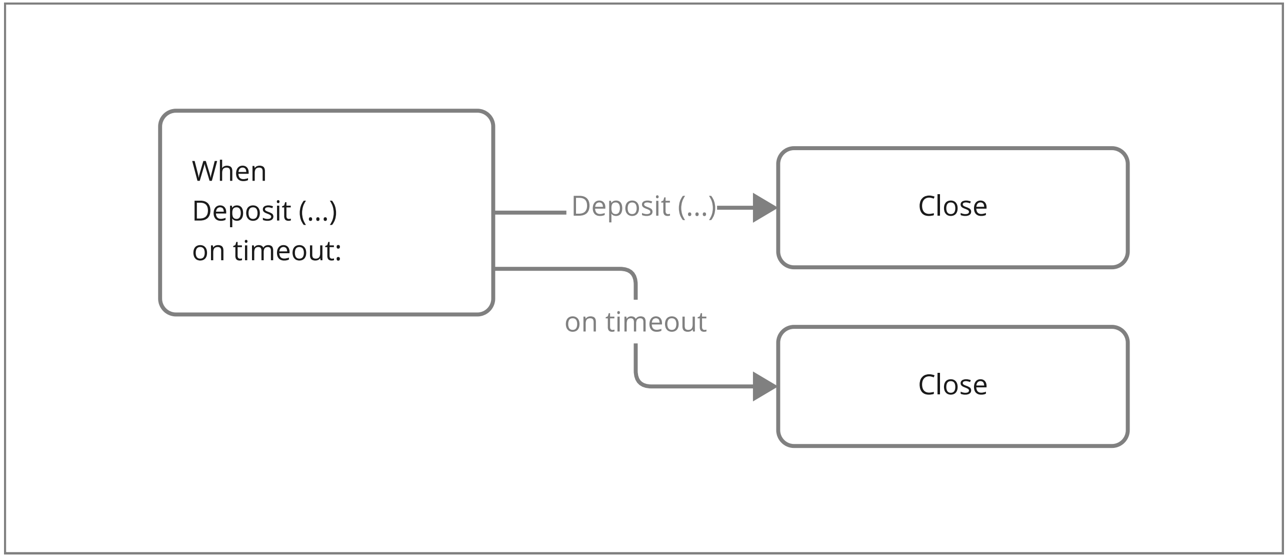
Exercise: "Buy Me a Coffee Request"

We want to make a "buy me a coffee" (donation) request from some address to your address for a 1000 Lovelace. We want to put it on the blockchain so the other party can send the money.

To do this:

- We will need your own wallet address.
- The wallet address of the donor.

Exercise: "Buy Me a Coffee Request"



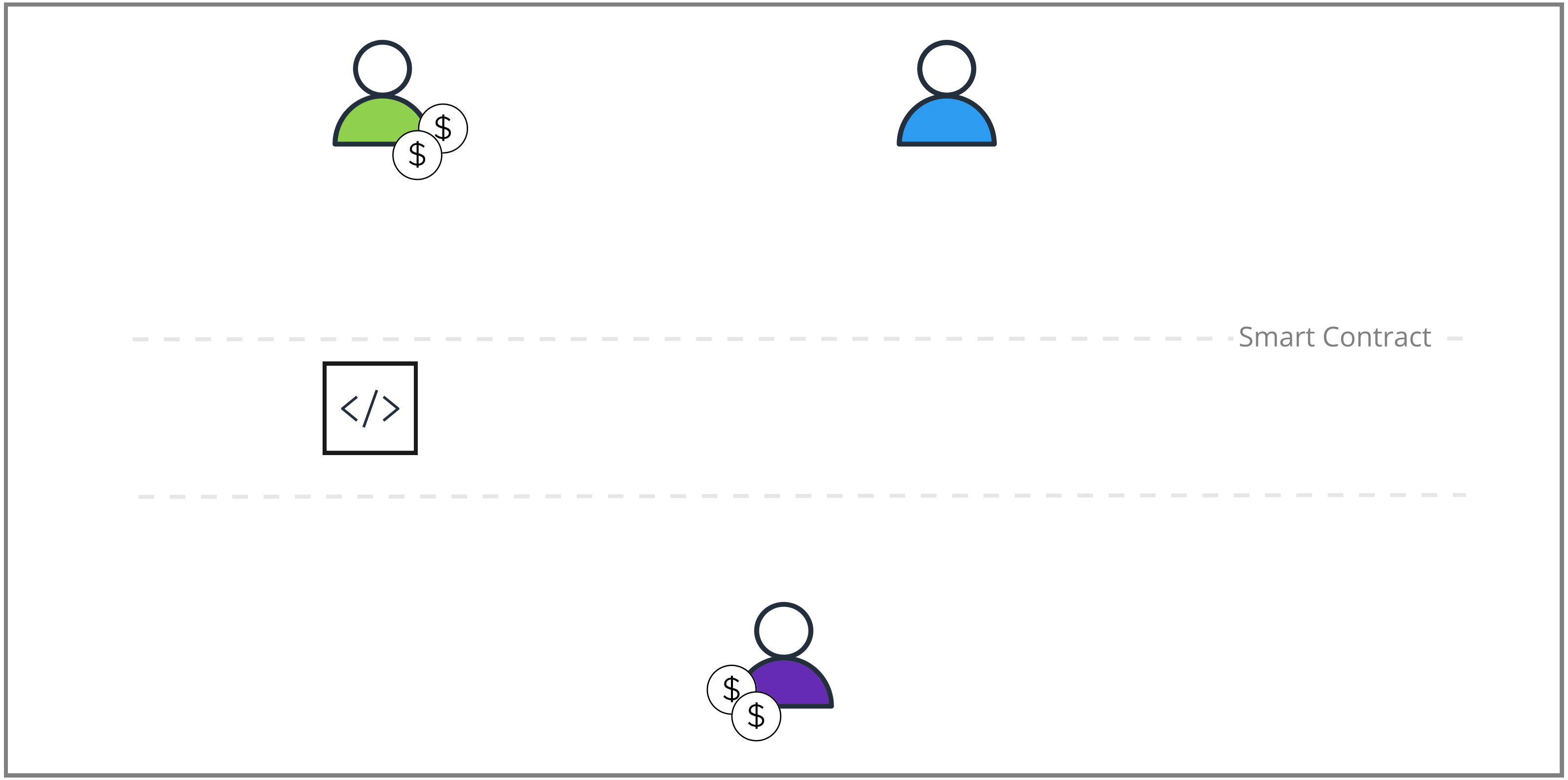
play.marlowe.iohk.io

preprod.marlowe.runner.iohk.io

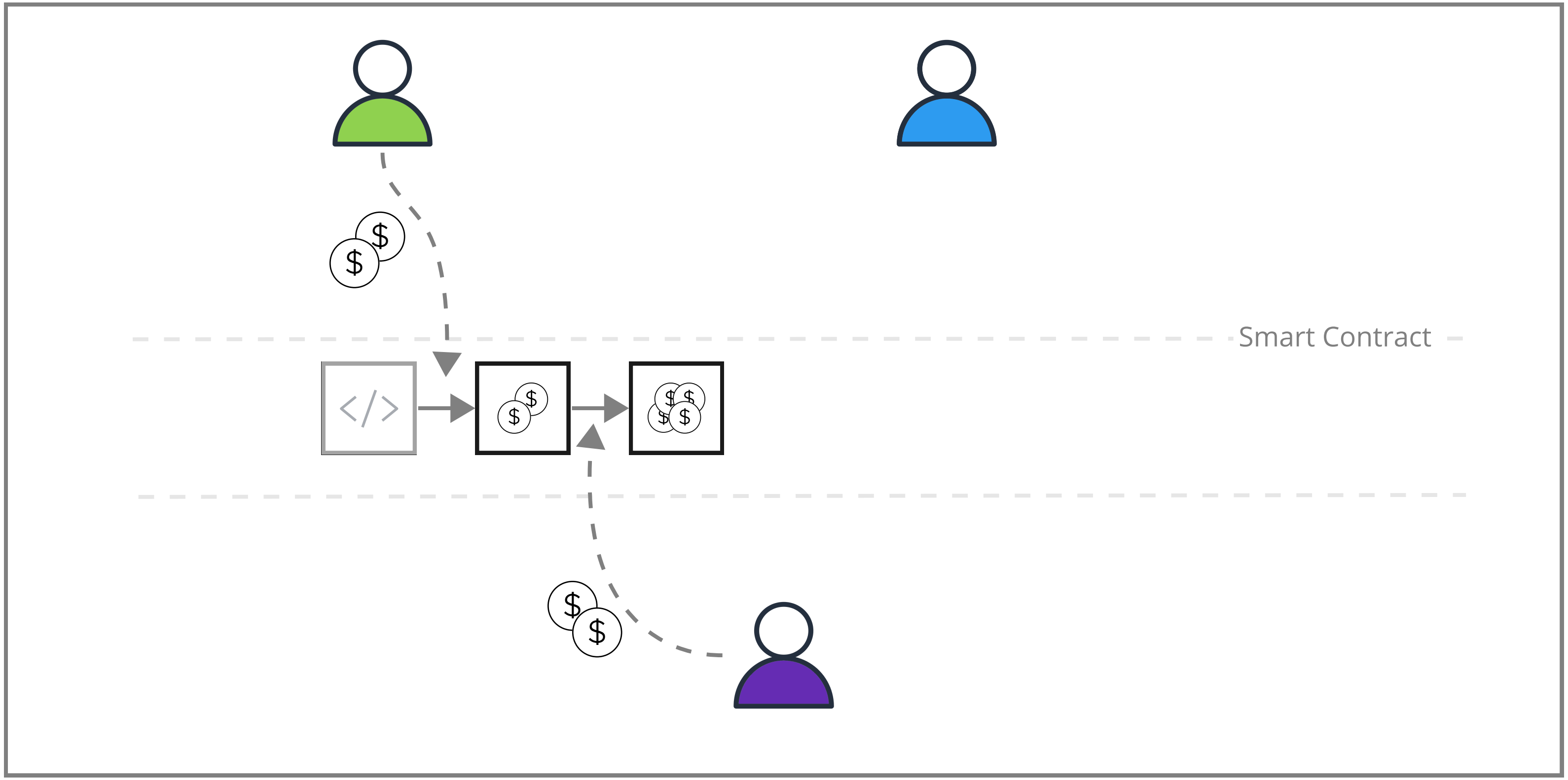
"Deposit" Locks The Assets

- Locked funds are controlled by the contract
- Depositor picks destination account

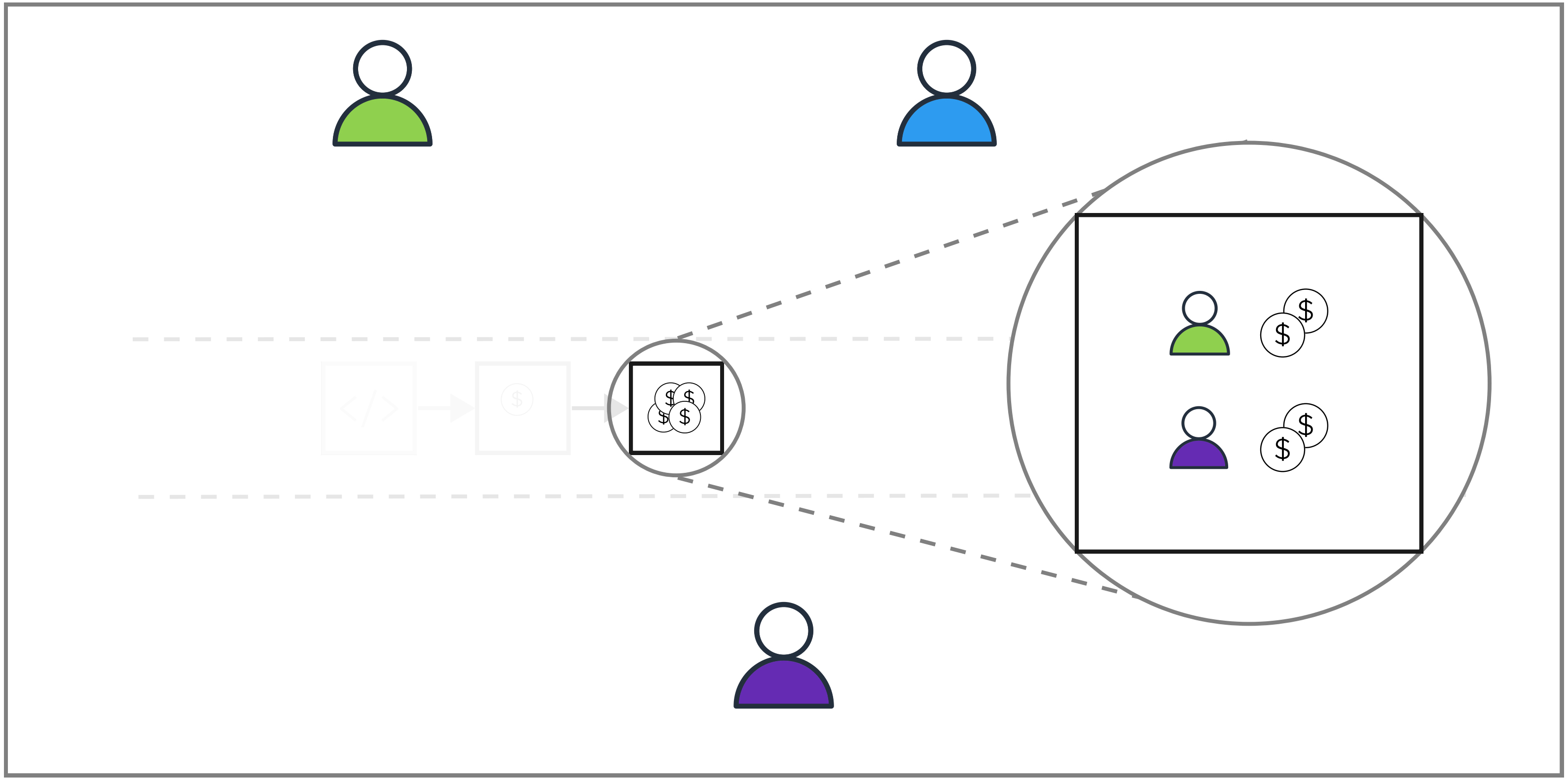
"Deposit" Locks The Assets



"Deposit" Locks The Assets



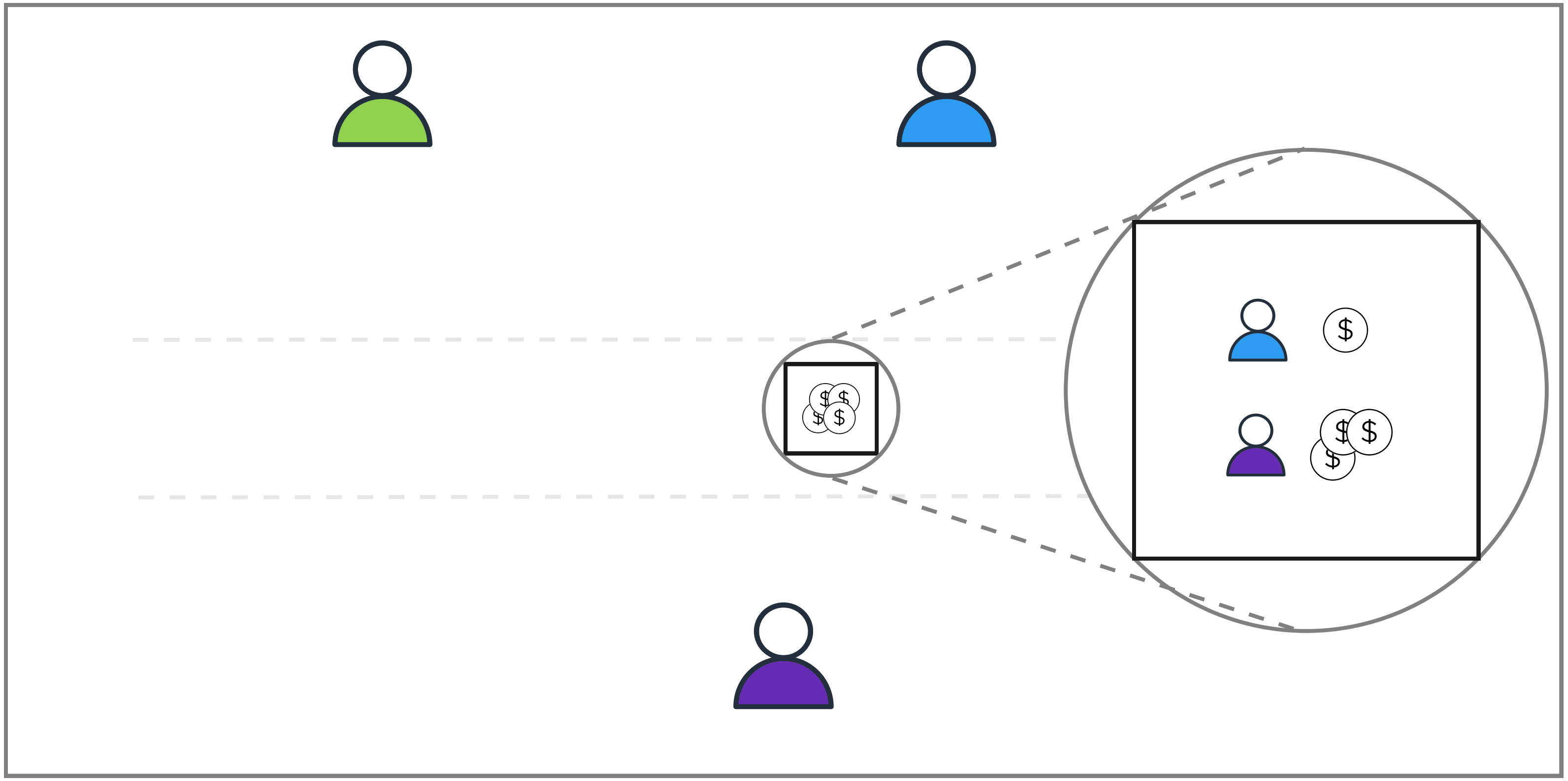
"Deposit" Locks The Assets



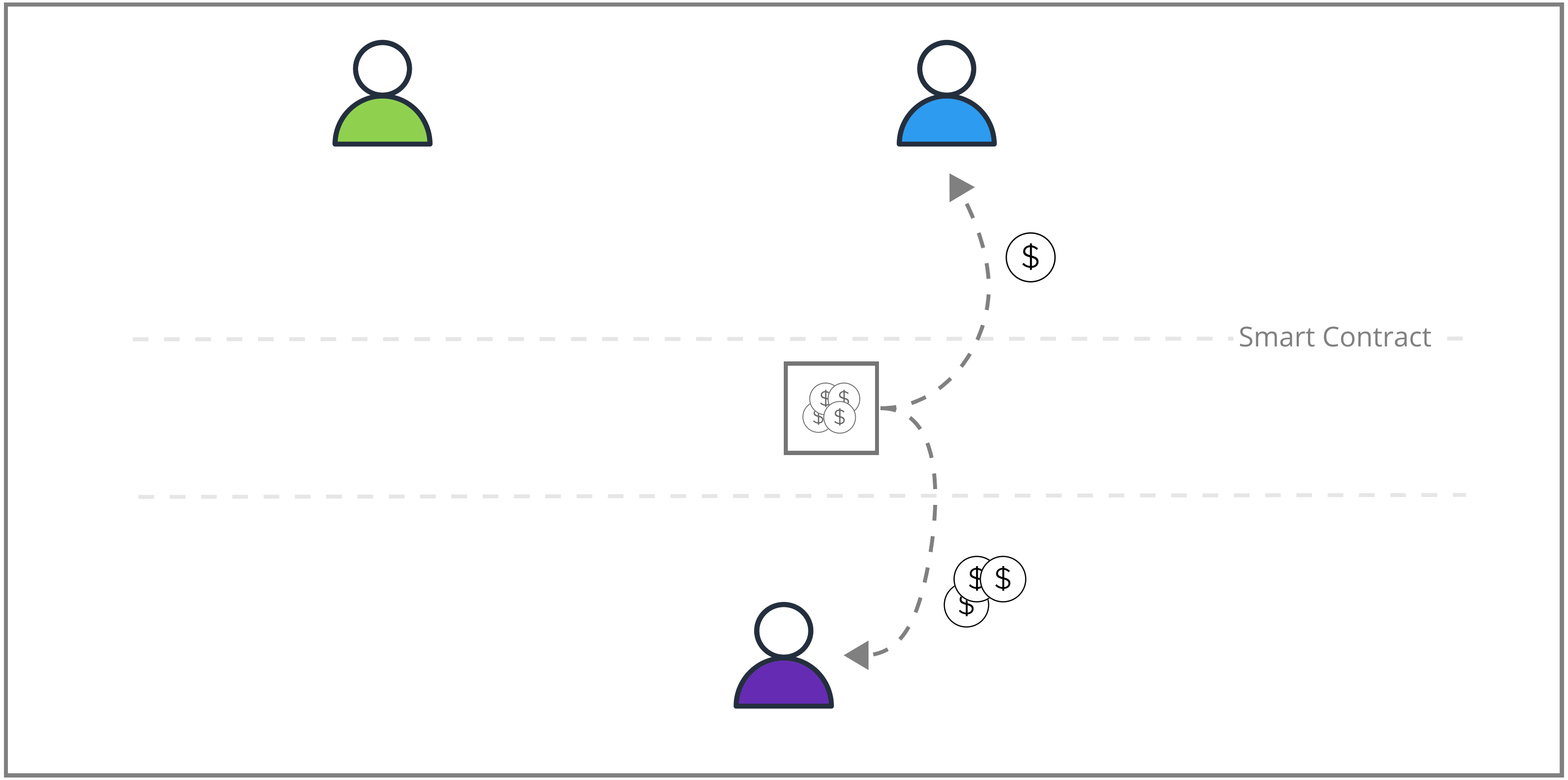
"Close" Releases The Assets

- When we reach "Close" contract releases remaining assets
- The assets are paid out according to account state

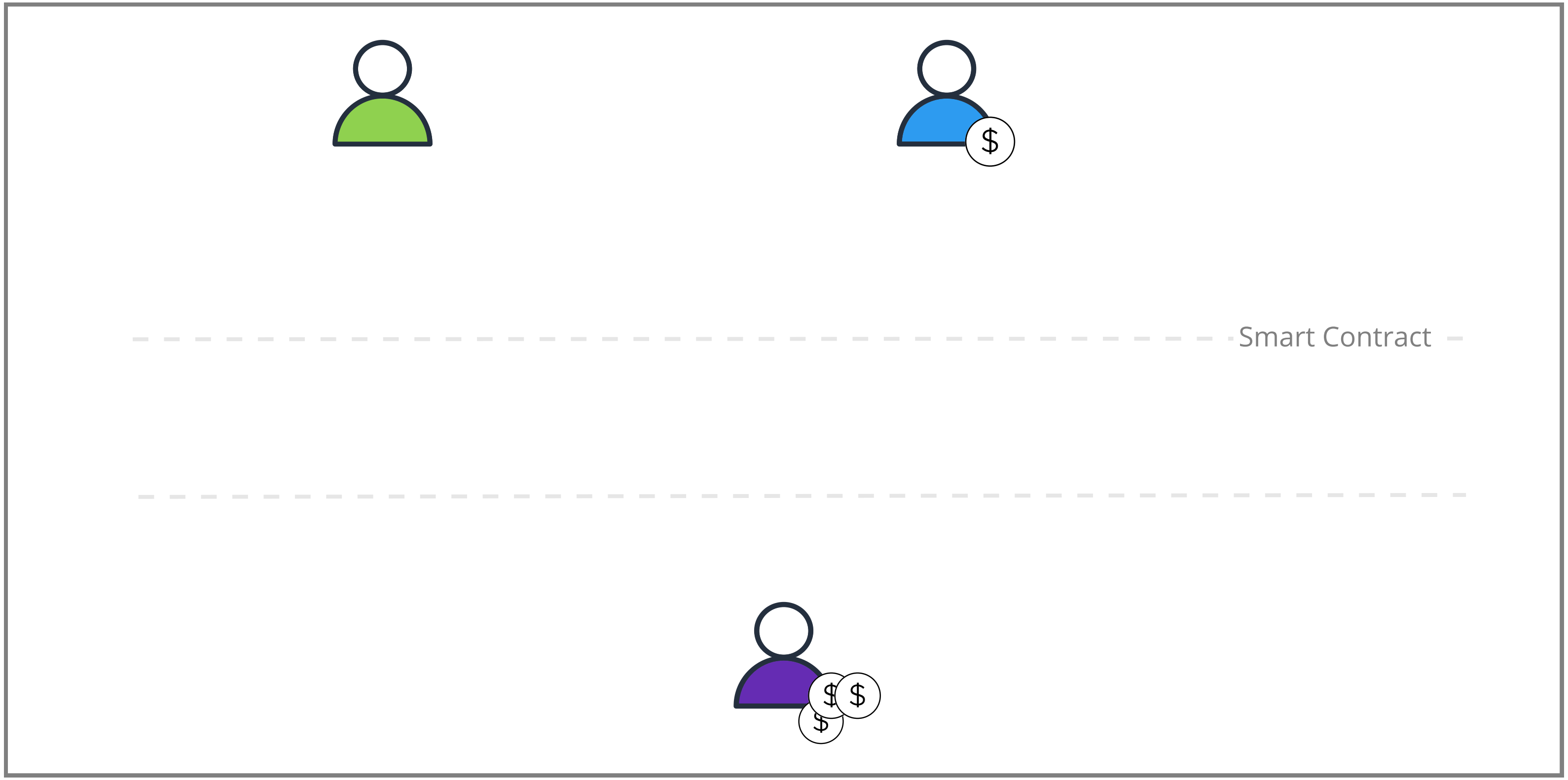
"Close" Releases The Assets



"Close" Releases The Assets



"Close" Releases The Assets



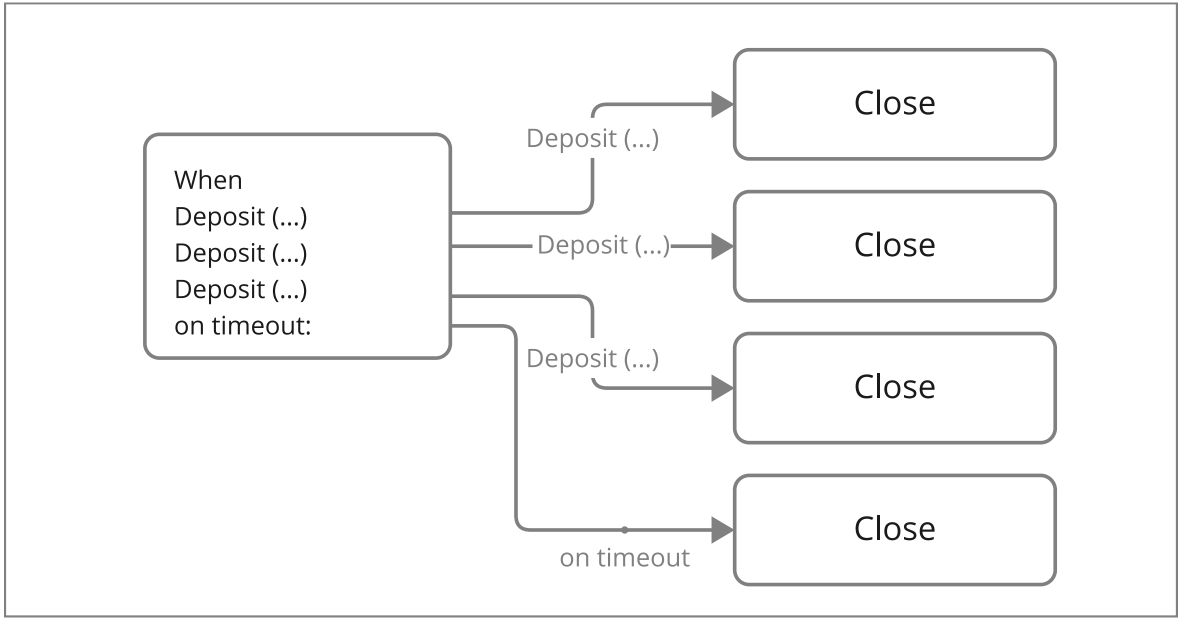
Exercise: "Coffee Plans"

This will be a variation of the previous exercise except that we give the donor three donation options:

- 1 000 000 Lovelace (1 ADA)
- 5 000 000 Lovelace (5 ADA)
- 10 000 000 Lovelace (10 ADA)

Please switch roles in pairs.

Exercise: "Coffee Plans"



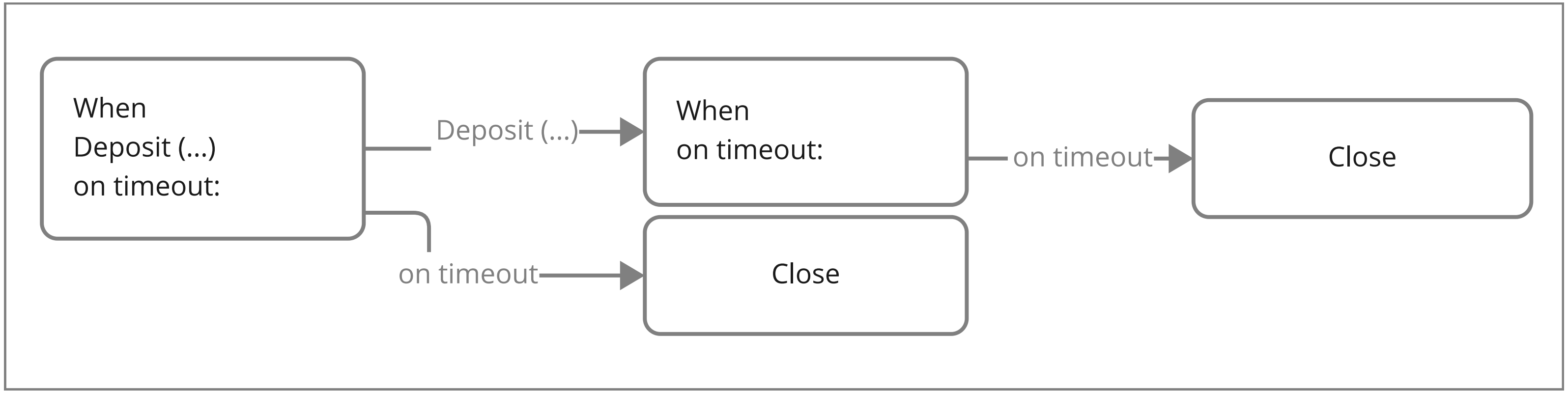
play.marlowe.iohk.io

preprod.marlowe.runner.iohk.io

Exercise: "Saving Account"

- We want to save some money for the future (because we spend too much ;-)
- We want to lock 1000 Lovelace in the contract.
- Let's lock it for testing for 3 minutes.
- After this period we would like to unlock the funds.

Exercise: "Saving Account"



play.marlowe.iohk.io

preprod.marlowe.runner.iohk.io

Plan

- How Marlowe Operates
- **The Developer Journey**
- Building DApps with Marlowe

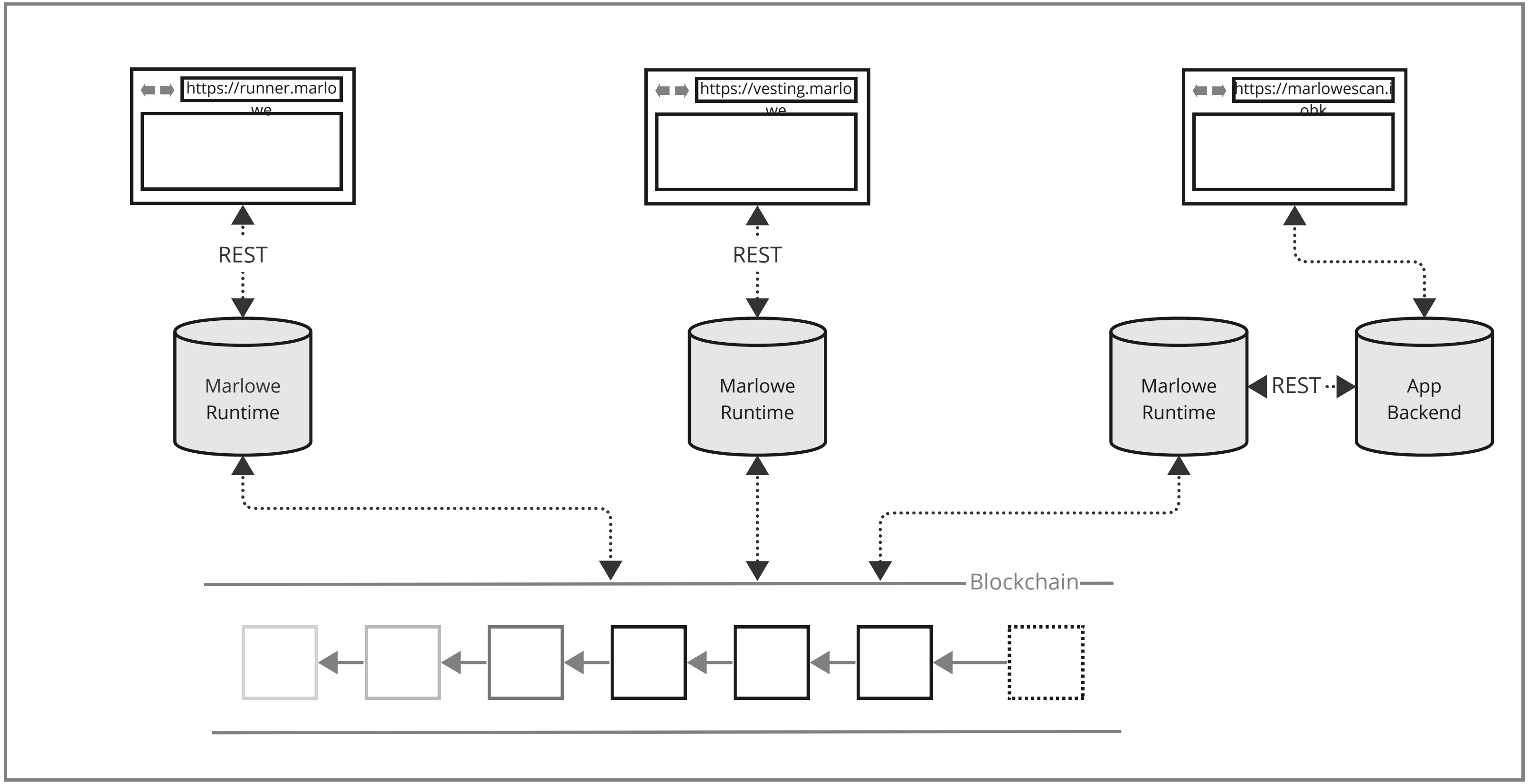
Plan

- How Marlowe Operates
- The Developer Journey
- **Building DApps with Marlowe**

DApps Prototypes

- Vesting app: vesting-preprod.scdev.aws.iohkdev.io
- Marlowe Runner: preprod.runner.marlowe.iohk.io

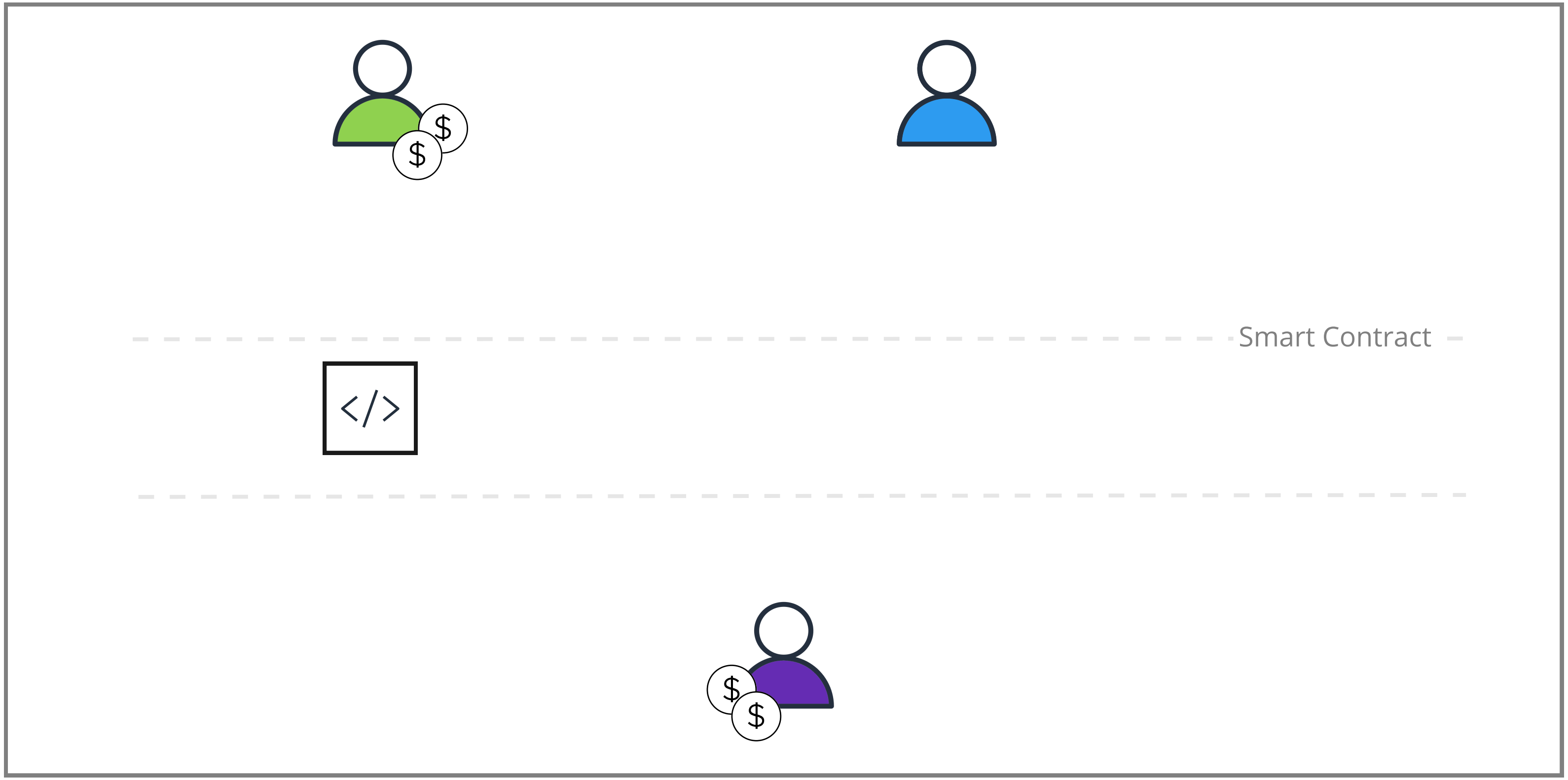
Marlowe Runtime



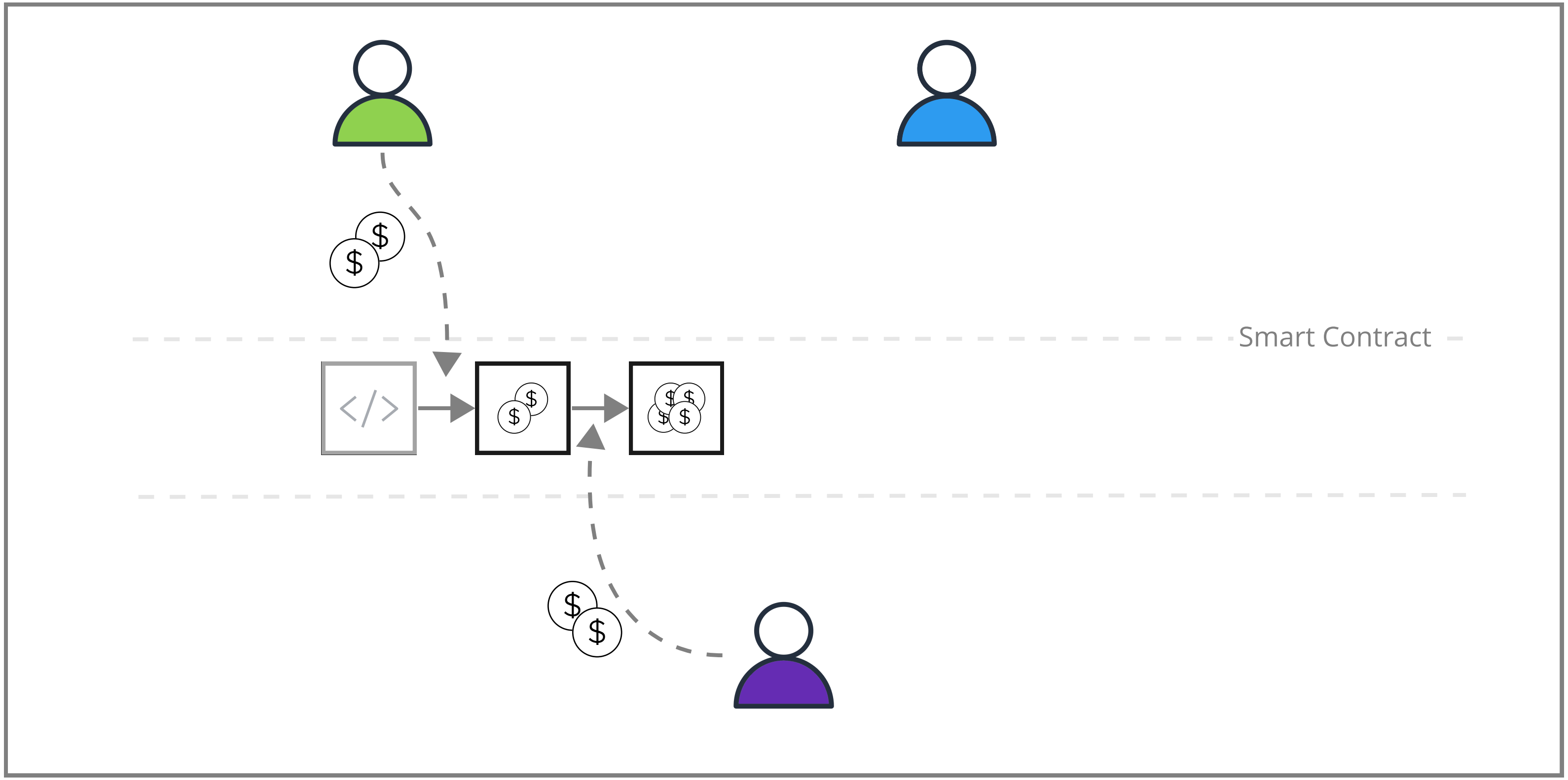
Marlowe and Your Dev Team

- Marlowe can be used from any programming language (on going integrations: Go, Rust, C#, Python, TypeScript,
- We are focusing on TypeScript/Javascript
- Marlowe Runtime is language agnostic (REST)

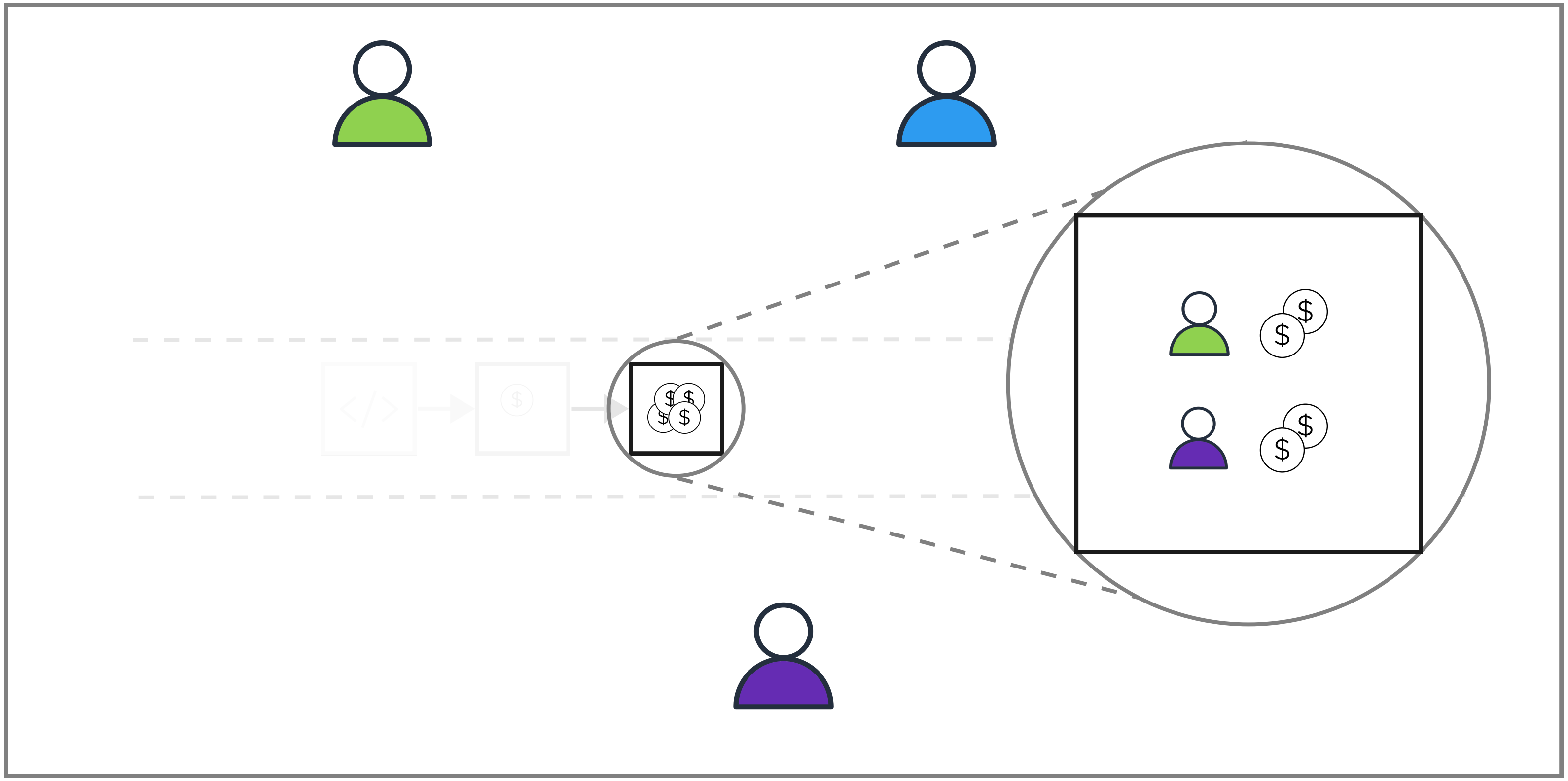
Example: "On-Chain Betting"



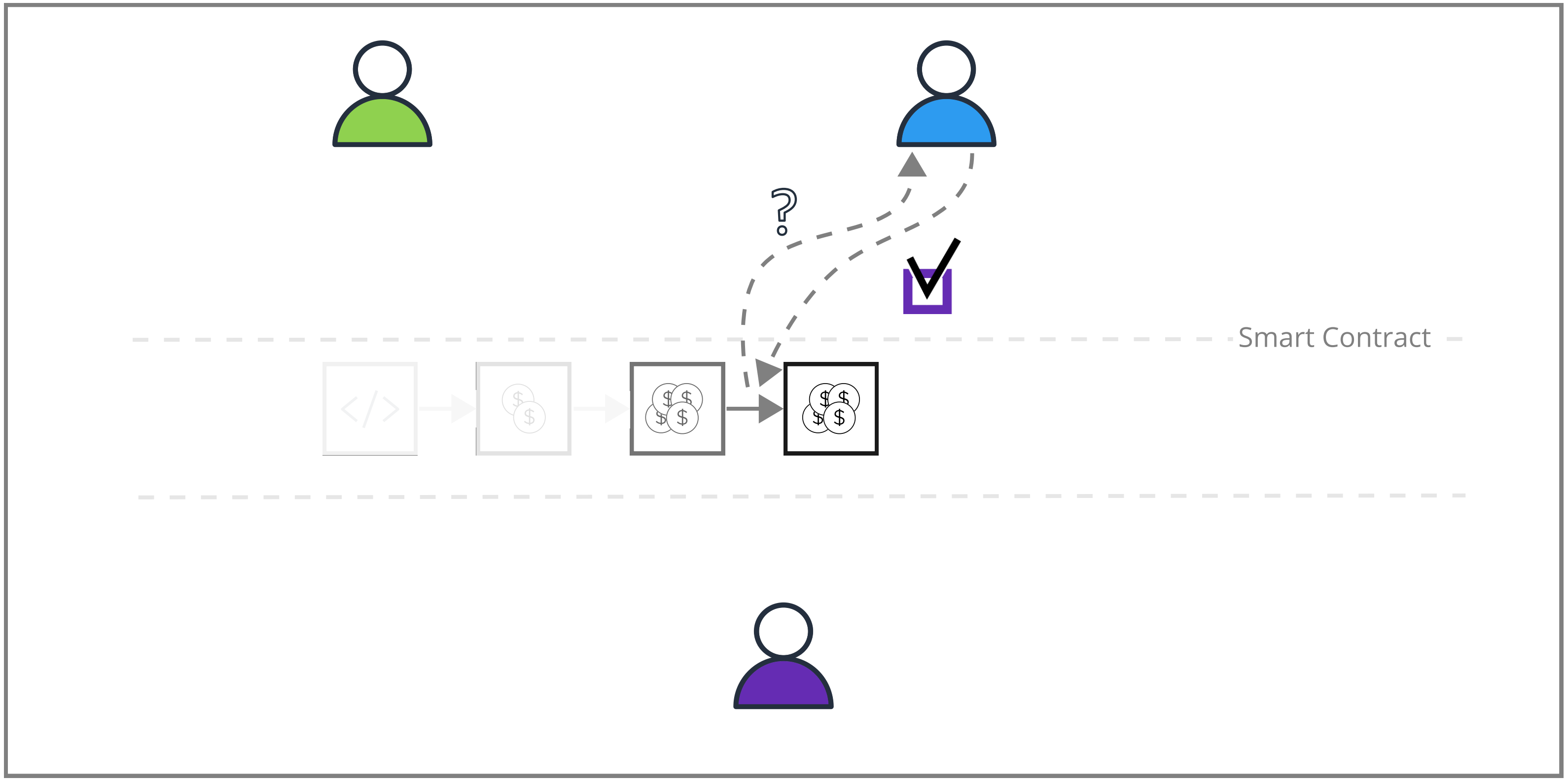
Example: "On-Chain Betting"



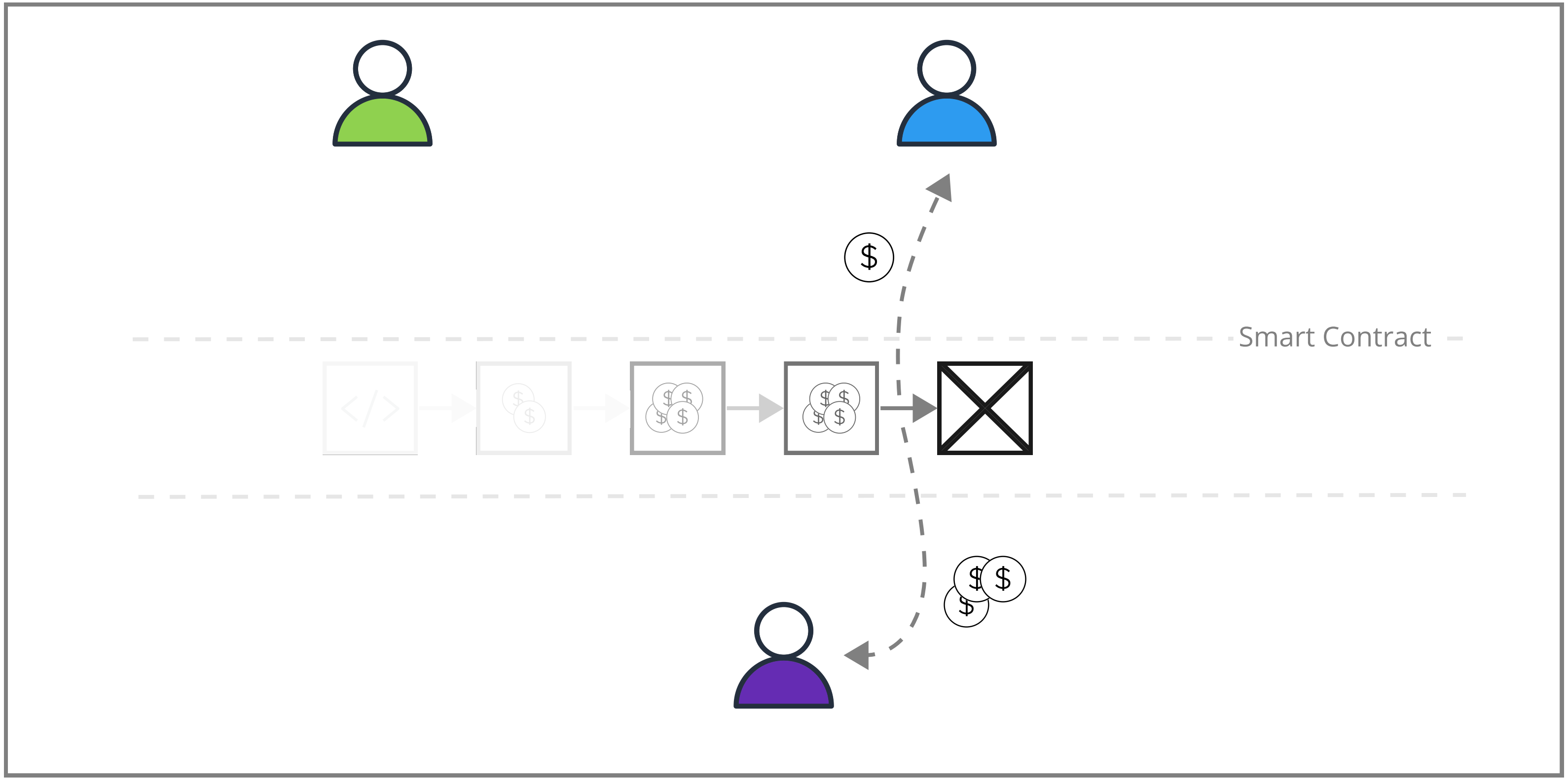
Example: "On-Chain Betting"



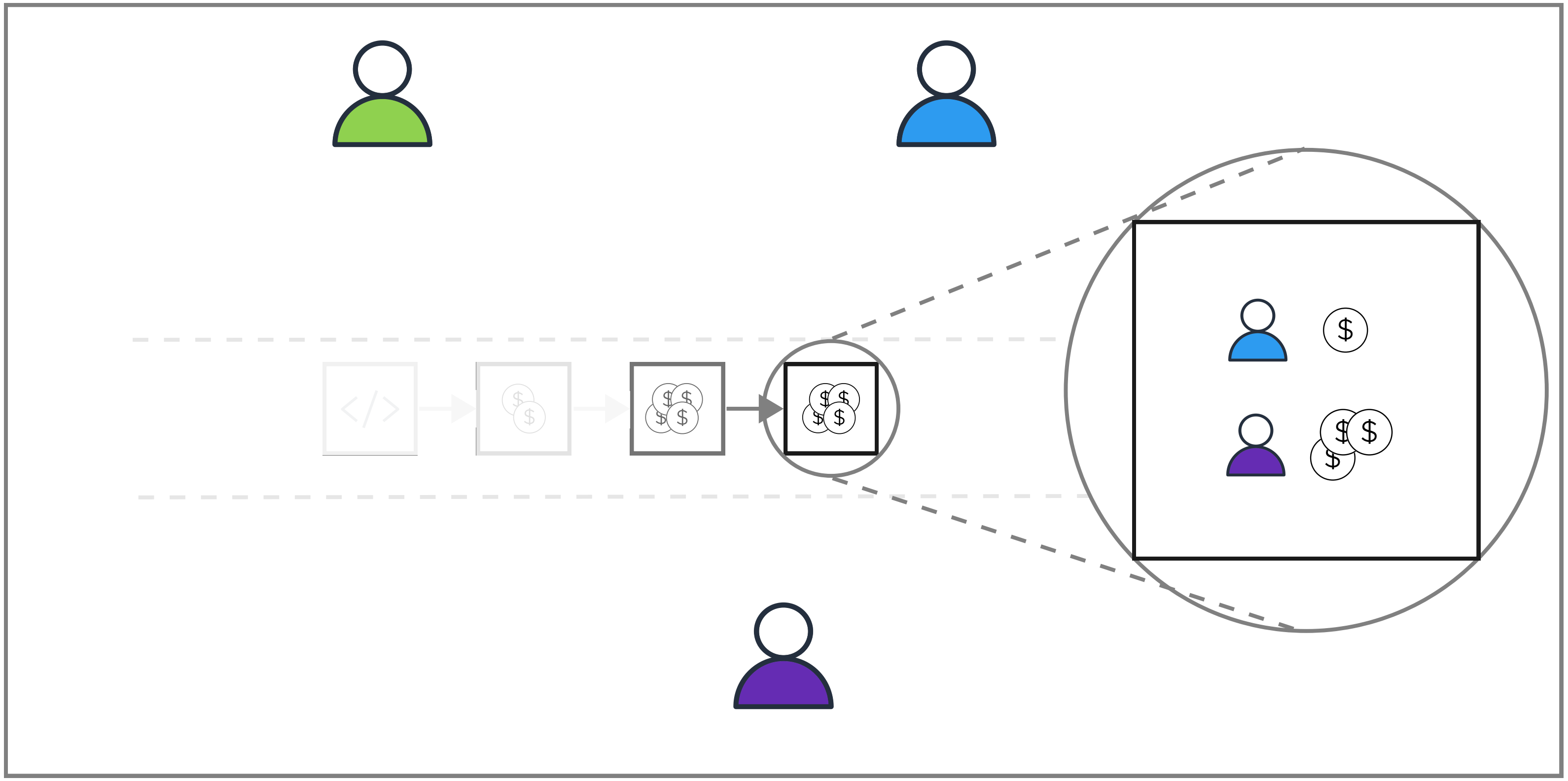
Example: "On-Chain Betting"



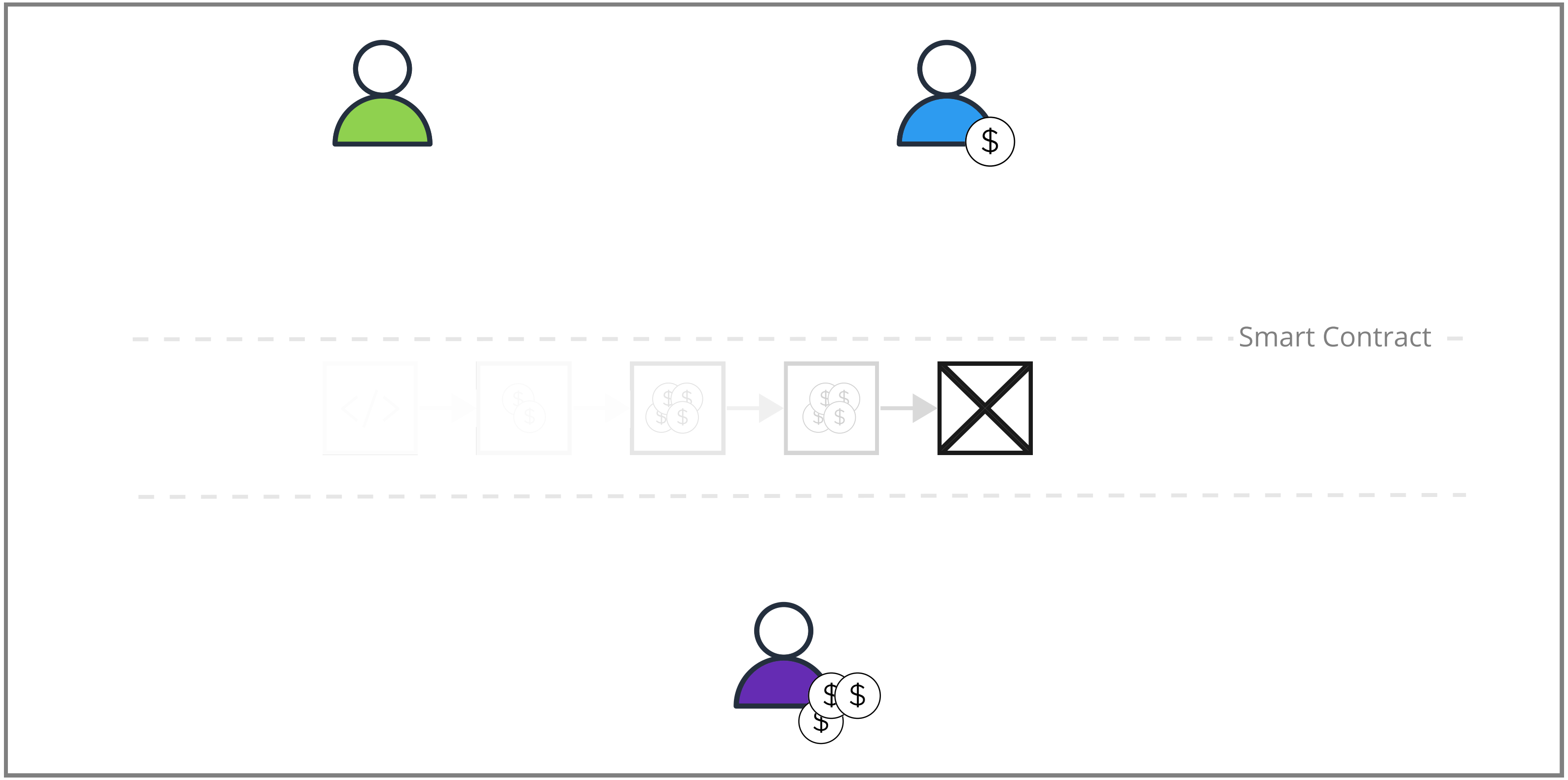
Example: "On-Chain Betting"



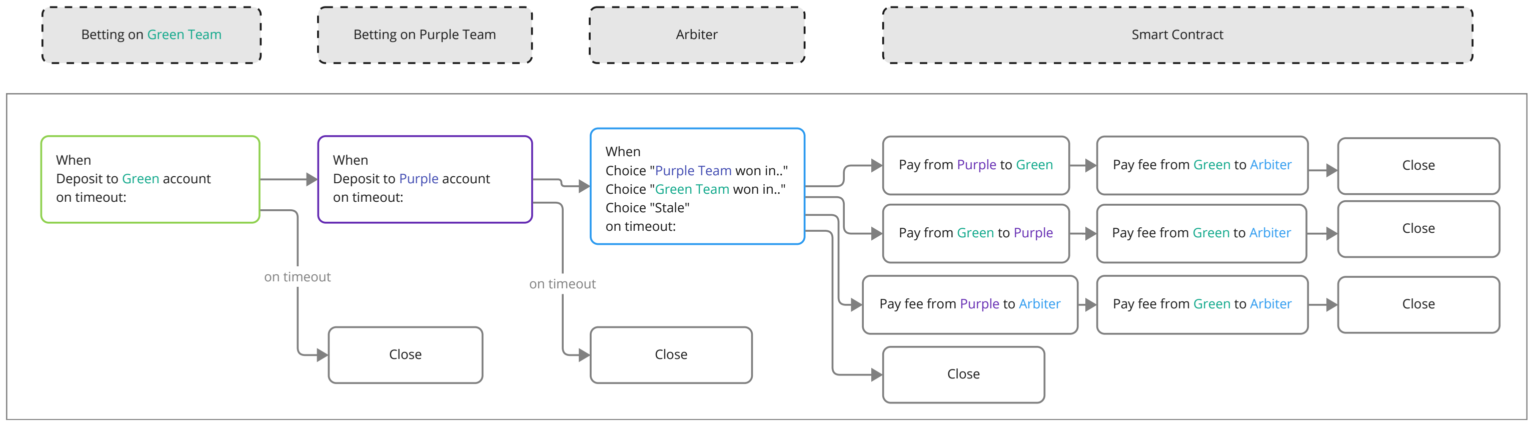
Example: "On-Chain Betting"



Example: "On-Chain Betting"



Example: "On-Chain Betting"



Let's analyse this contract in play.marlowe.iohk.io

On-Chain Betting Benefits

- No up front deposits / account charging etc.
- Predictable cash flow
- Global reaching service
- Everything is transparent
- Arbiter has reputation at stake

Thank you

Exercise: "Donation Request"

We want to make a donation request from your partner for a 1000 Lovelace. We want to put it on the blockchain so the other party can send the money.

To do this:

- We will need your own wallet address.
- The wallet address of the donor.

Exercise: "Donation Request"

https://input-output-hk.github.io/marlowe-ts-sdk/modules/_marlowe_io_language_core_v1.index.html

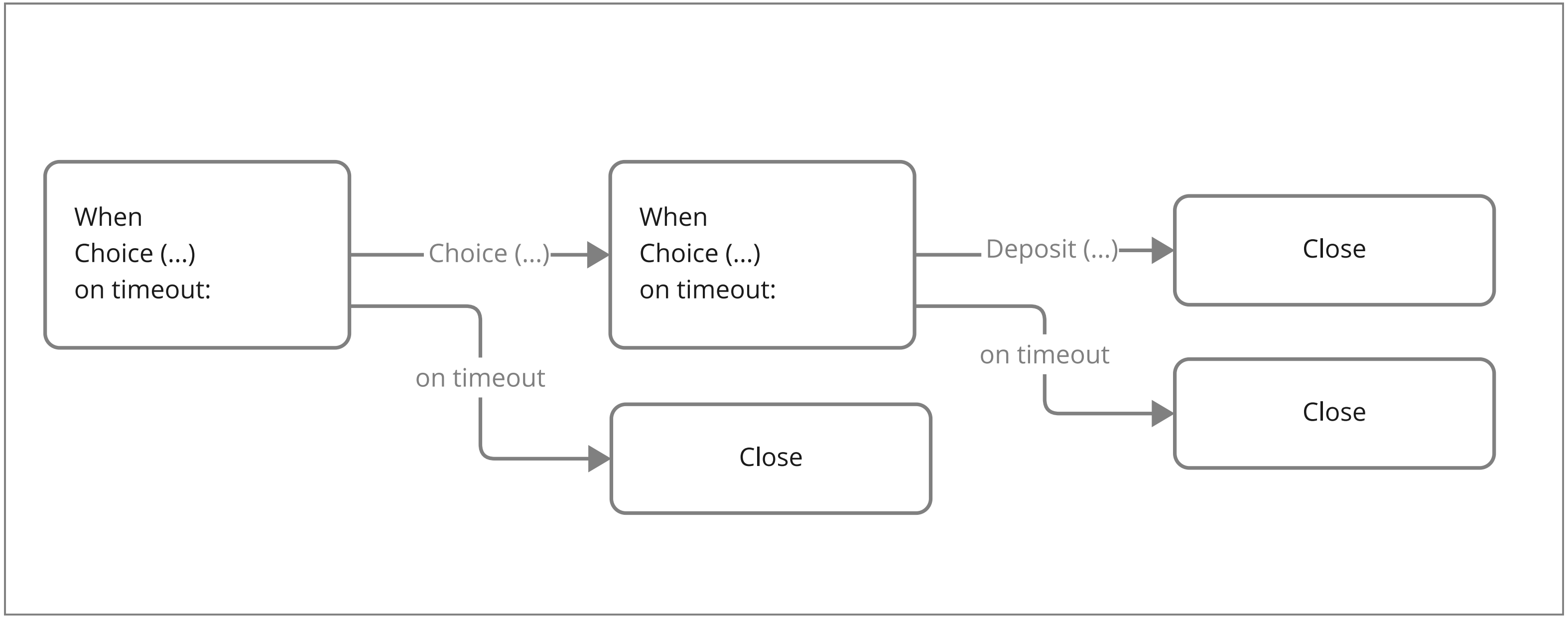
https://input-output-hk.github.io/marlowe-ts-sdk/modules/_marlowe_io_wallet.html

Exercise: "Flex Coffee Donation"

This will be a variation of the previous exercise except that we give the donor full flexibility to pick a value between:

1 000 000 - 10 000 000 Lovelace

Exercise: "Flex Coffee Donation"



play.marlowe.iohk.io

preprod.marlowe.runner.iohk.io