Solidity

Pavel Lukačik

Solidity (.sol)

- objektovo orientovaný programovací jazyk určený pre vytváranie smart kontraktov na platforme Ethereum
- ovplyvnený C++, Python a JavaScript a dizajnovaný pre Ethereum Virtual Machine (EVM)
- staticky typovaný, podporuje dedenie, knižnice, ...
- aktuálne v0.7.5

Smart contract

- program, alebo transakčný protokol, ktorý zaisťuje, overuje alebo vynucuje vyjednanie či uskutočnenie kontraktu (dohody)
- účelom je zbaviť sa potreby sprostredkovateľov, zbytočných poplatkov, strát kvôli podvodom, ...
- aplikácie pre finančný obchod, poistenie, vlastníctvo majetku, pôžičky, hlasovanie (voľby), ...

Ethereum

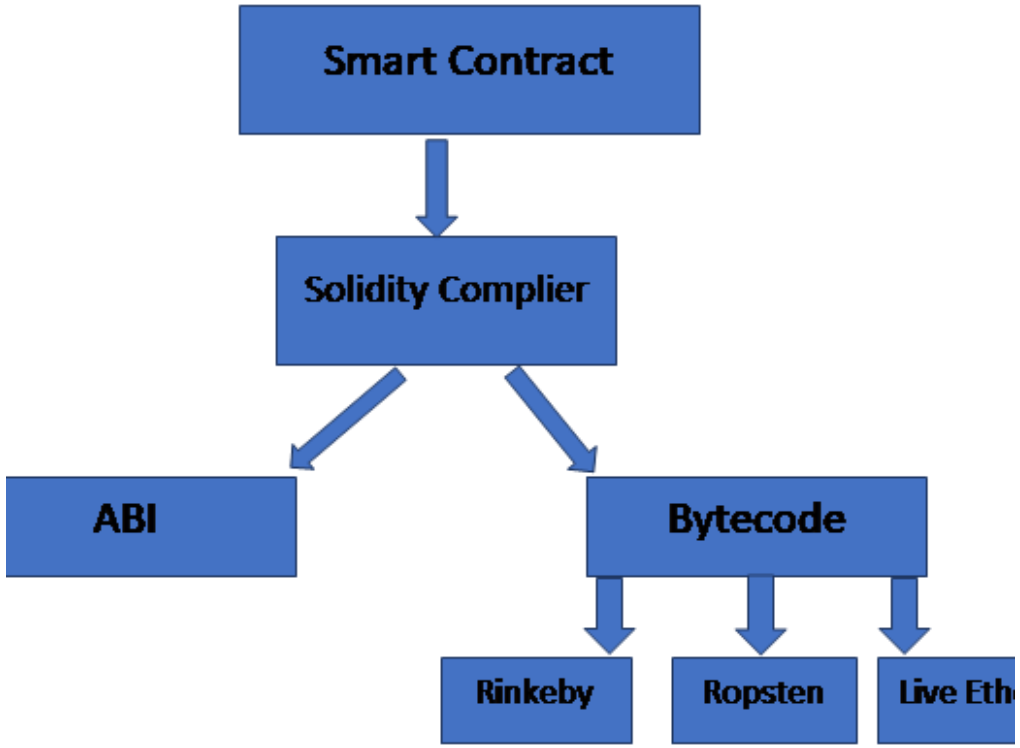
- open source platforma založená na decentralizovanej databáze, tzv blockchain s funkcionalitou smart kontraktov
- Ether (ETH) – kryptomena
- slúži ako platforma pre viac ako 1 900 rôznych kryptomien (tokenov)
- autorom je Vitalik Buterin (2013)

Blockchain (block chain)

- decentralizovaná databáza
- stále rastúci list záznamov
- chránené proti neoprávnenému zásahu z vonkajšej strany, ale aj zo strany samotného uzla peer-to-peer siete
- najčastejšie použitie – účtovná kniha (ledger) krypomien
- ukážka... (https://andersbrownworth.com/blockchain/)

Jednoduchý kontrakt („Hello World“)

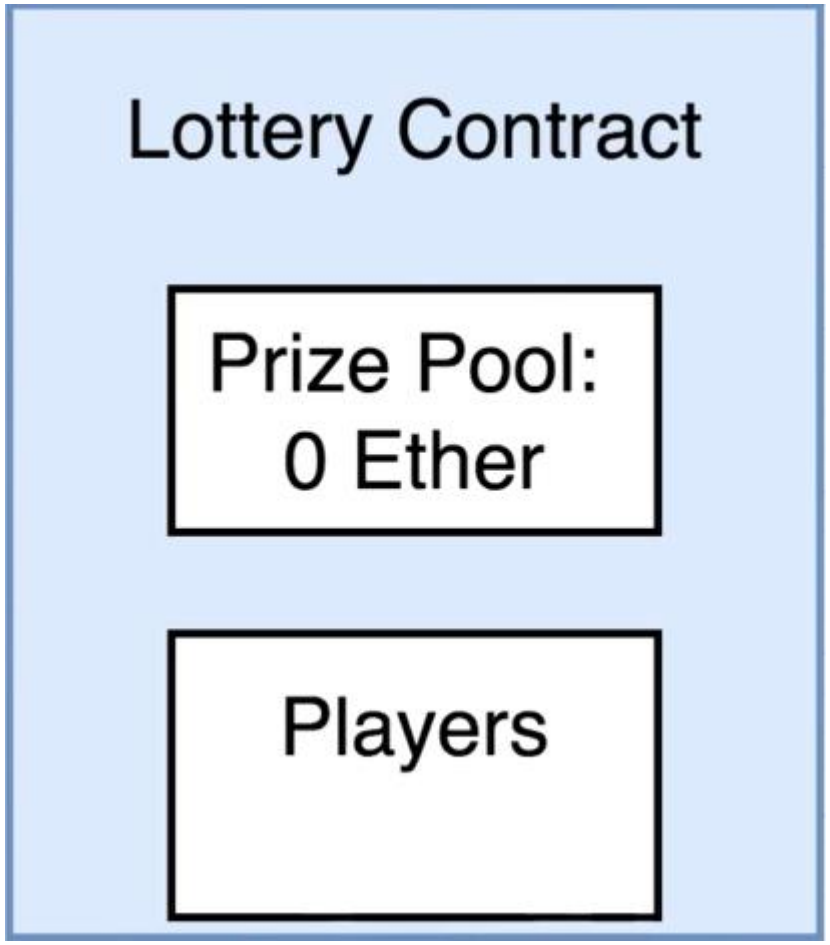
- kód = definícia kontraktu

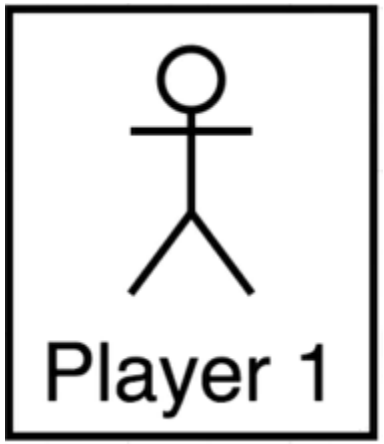


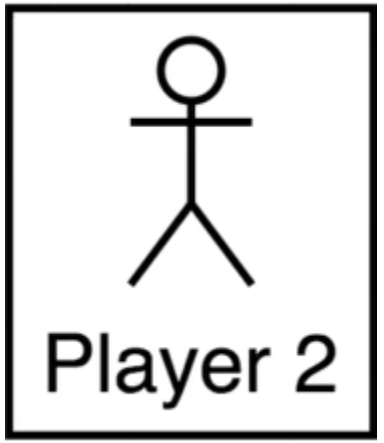
- http://remix.ethereum.org/

Running Contract Functions	
'Calling' a Function	Sending a Transaction to a Function
Cannot modify the contract's data	Can modify a contract's data
Can return data	Takes time to execute!
Runs instantly	Returns the transaction hash
Free to do!	Costs money!

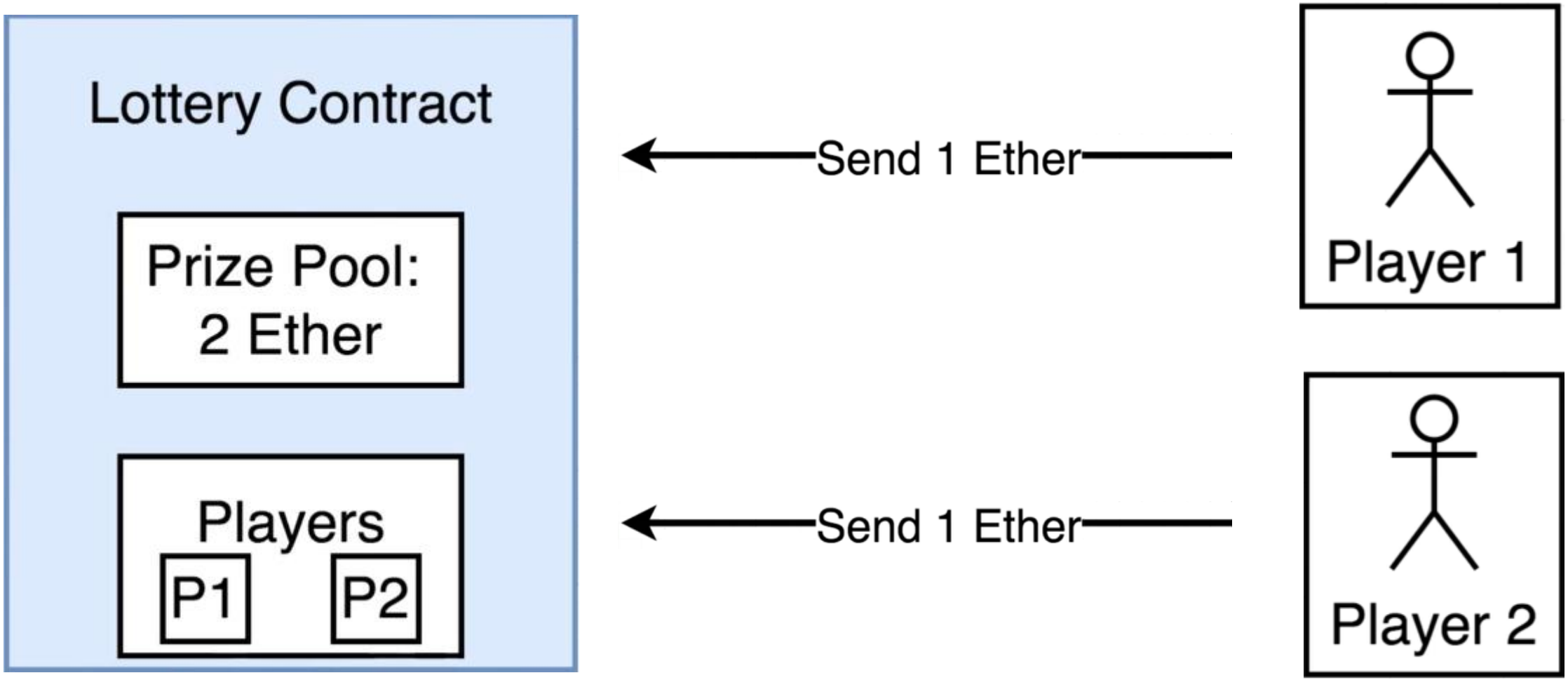
Lotéria


Lottery Contract
Prize Pool:
0 Ether
Players

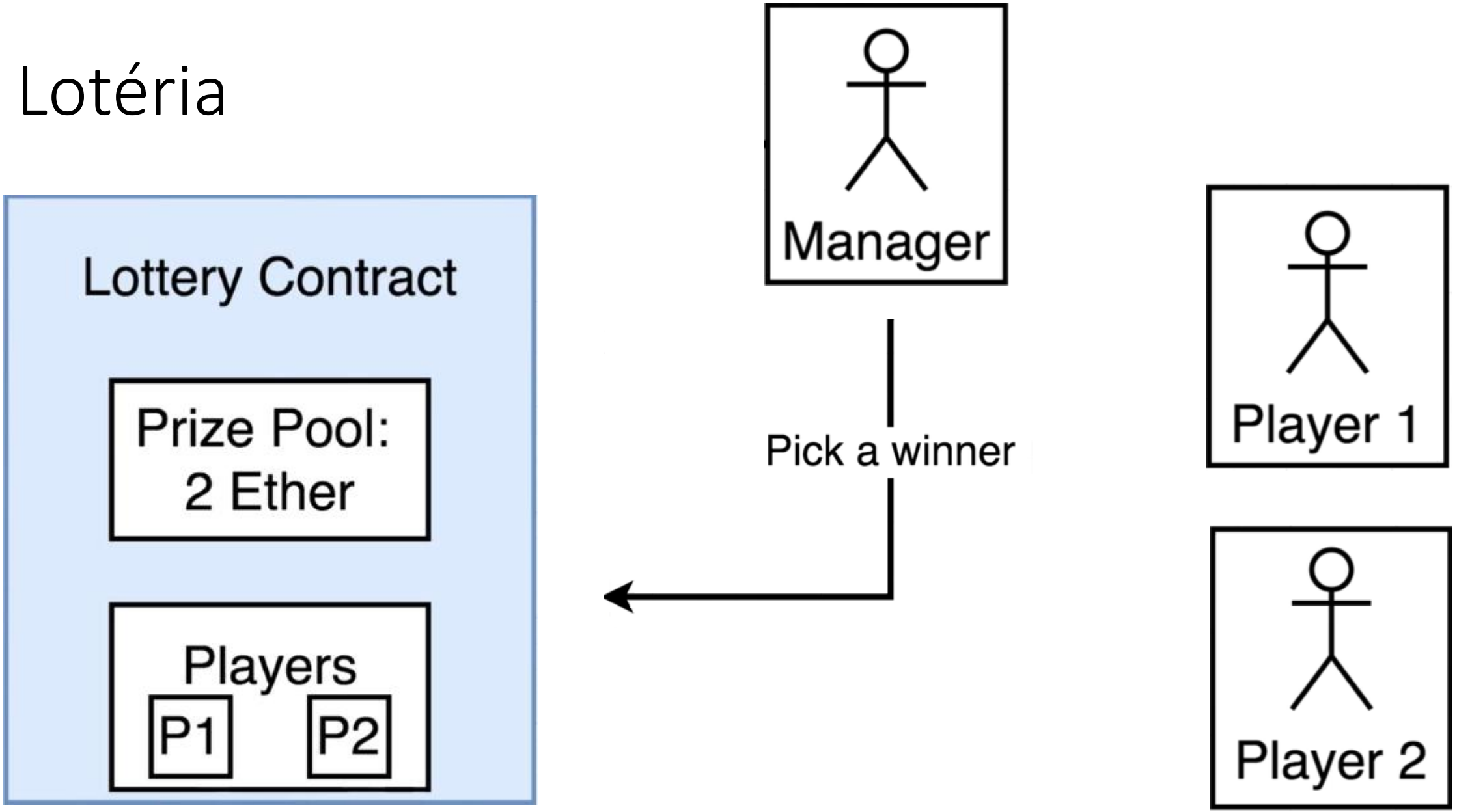

Player 1


Player 2

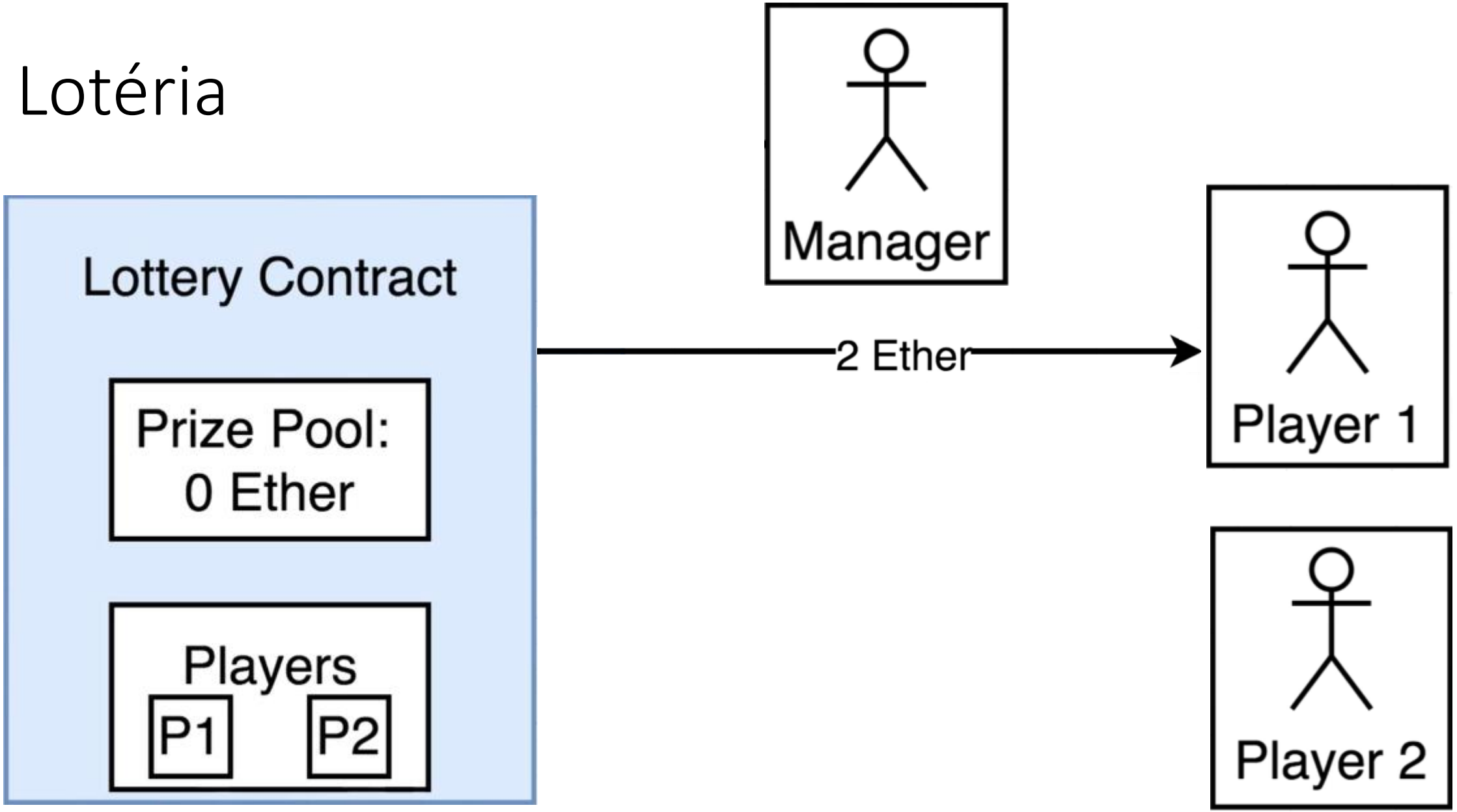
Lotéria


Lottery Contract
Prize Pool:
2 Ether
Players
P1
P2
Send 1 Ether
Send 1 Ether
Player 1
Player 2

Lotéria


Lottery Contract
Prize Pool:
2 Ether
Players
P1
P2
Manager
Pick a winner
Player 1
Player 2

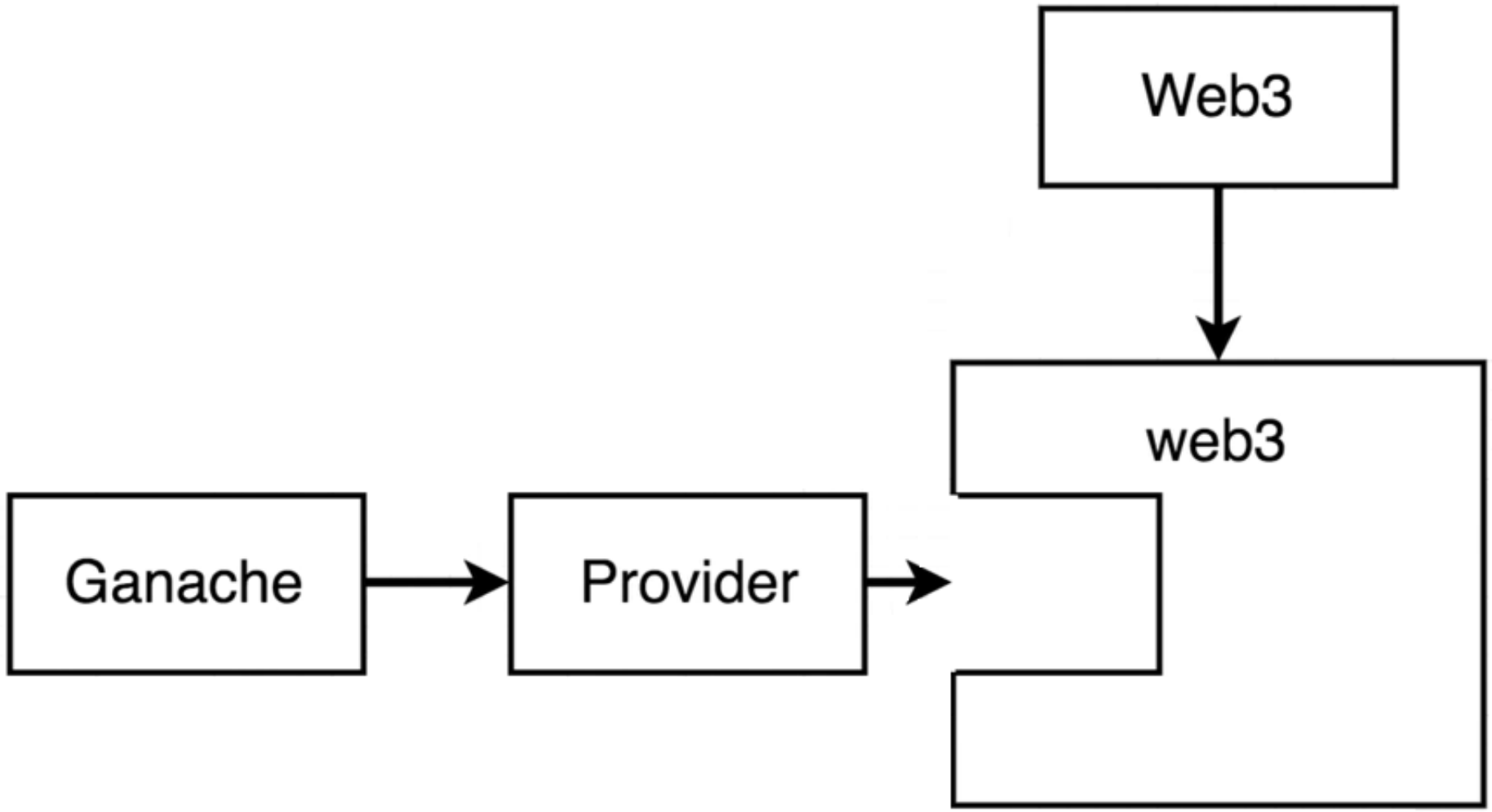
Lotéria


Lottery Contract
Prize Pool:
0 Ether
Players
P1
P2
Manager
2 Ether
Player 1
Player 2

The 'msg' Global Variable	
Property Name	Property Name
msg.data	'Data' field from the call or transaction that invoked the current function
msg.gas	Amount of gas the current function invocation has available
msg.sender	Address of the account that started the current function invocation
msg.value	Amount of ether (in wei) that was sent along with the function invocation

Basic Types		
Name	Notes	Examples
string	Sequence of characters	"Hi there!" "Chocolate"
bool	Boolean value	true false
int	Integer, positive or negative. Has no decimal	0 -30000 59158
uint	'Unsigned' integer, positive number. Has no decimal	0 30000 999910
fixed/ufixed	'Fixed' point number. Number with a decimal after it	20.001 -42.4242 3.14
address	Has methods tied to it for sending money	0x18bae199c8dbae199c8d

Reference Types		
Name	Notes	Examples
fixed array	Array that contains a *single type* of element. Has an unchanging length	`int[3] --> [1, 2, 3]` `bool[2] --> [true, false]`
dynamic array	Array that contains a *single type* of element. Can change in size over time	`int[] --> [1,2,3]` `bool[] --> [true, false]`
mapping	Collection of key value pairs. Think of Javascript objects, Ruby hashes, or Python dictionary. All keys must be of the same type, and all values must be of the same type	`mapping(string => string)` `mapping(int => bool)`
struct	Collection of key value pairs that can have different types.	`struct Car { string make; string model; uint value; }`


Web3
web3
Ganache
Provider

Ďakujem za pozornosť

https://s.ics.upjs.sk/~plukacik/SPS2020