

Privacy in Smart-Contract based fair exchanges

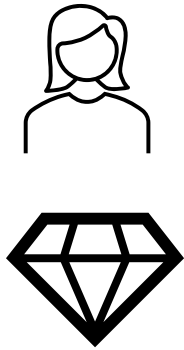
Preston Haffey

University of Calgary MSc. Computer Science

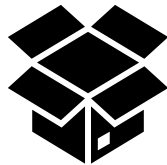
Plan:

- Basic Exchange
- Fairness in exchange
- Exchange digital item for digital coins
- Blockchains
- Smart contracts
- Smart-contract based fair exchanges
- Privacy during disputes
- Providing privacy in disputes

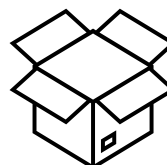
An Exchange

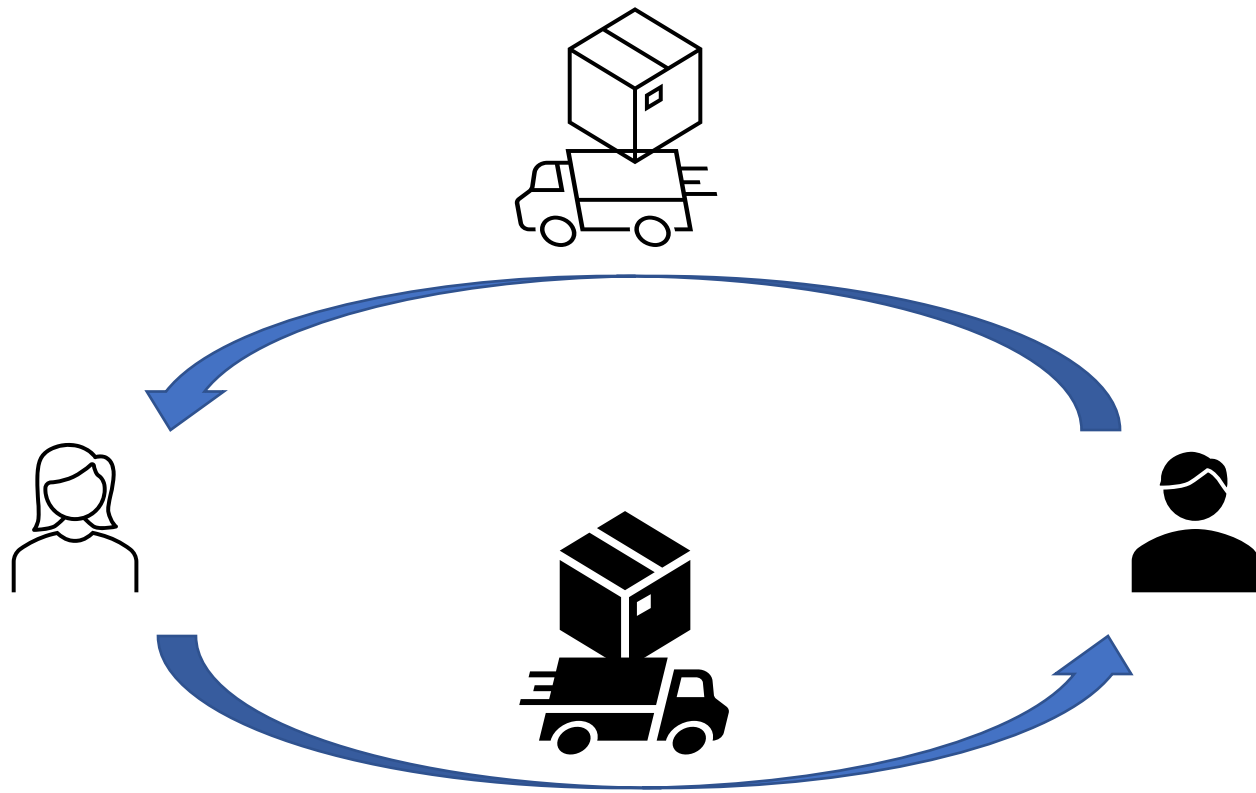


I'll trade you
Diamond for
Gold

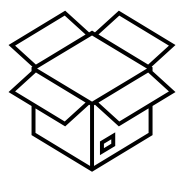


I'll accept
Diamond in
exchange for
Gold

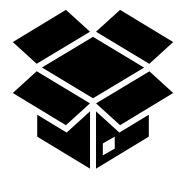




Thanks!

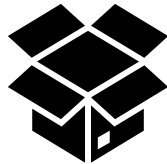
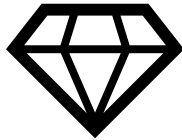
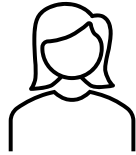


Thank you!

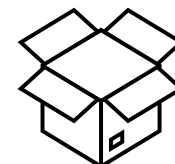


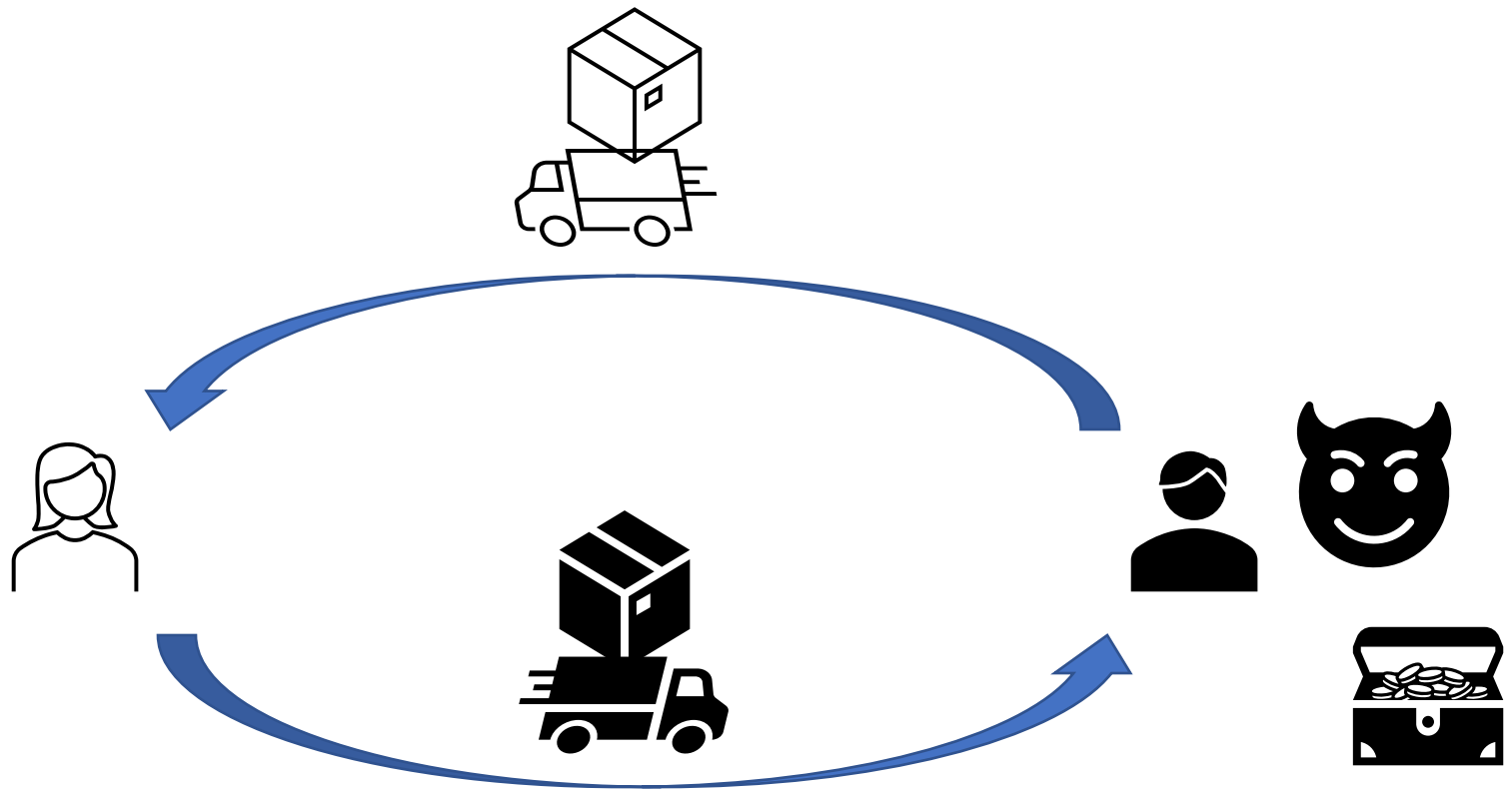
But what can go wrong?

I'll trade you
Diamonds for
Gold

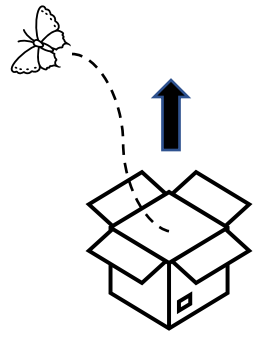


I'll accept
Diamond in
exchange for
Gold

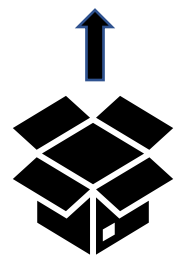
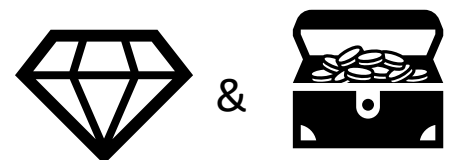
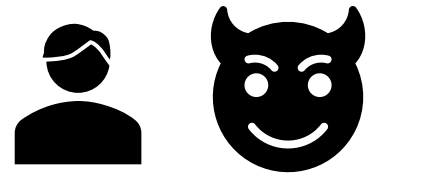




This box is empty!

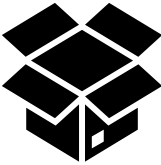
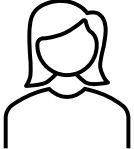


Haha, thanks for the Diamond!

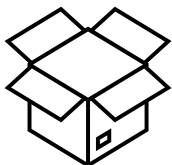


An Exchange: What else can we try?

Send Gold first
and Diamond
will follow



You should
send first!

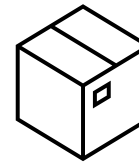


Without trust, we've reached an impasse

Alice and Bob want a **fair** exchange where Alice and Bob are guaranteed to receive exactly what they wanted or lose nothing.



But how can this be achieved?



It has been proven that two-party fair exchange is impossible to achieve without a Trusted Third Party (TTP).



Cleve, R.: Limits on the security of coin flips when half the processors are faulty. In: Proceedings of the eighteenth annual ACM symposium on Theory of computing. pp. 364–369 (1986)

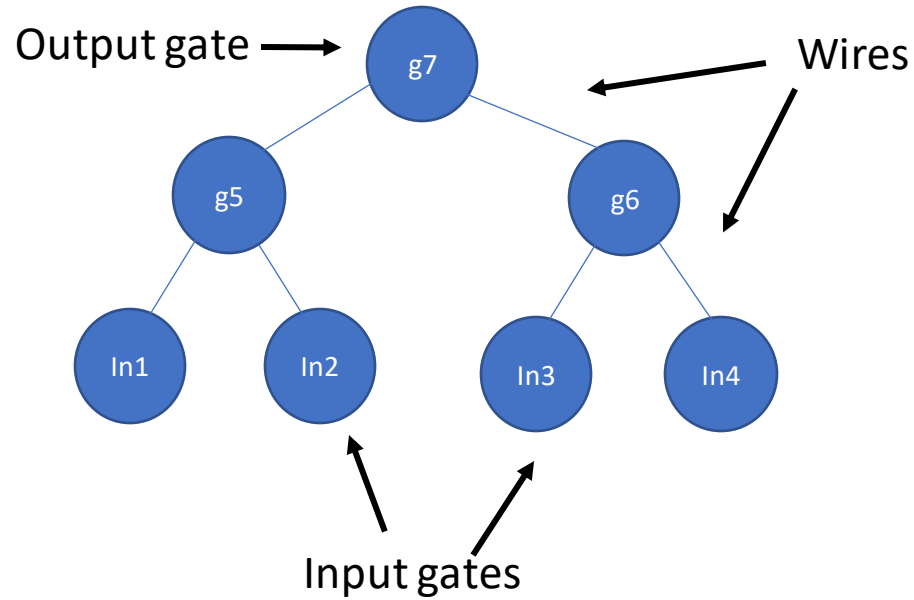
Garbinato, B., Rickebusch, I.: Impossibility results on fair exchange. 10th International Conference on Innovative Internet Community Systems (I2CS)–Jubilee Edition 2010– (2010)

Pagnia, H., Gärtner, F.C.: On the impossibility of fair exchange without a trusted third party. Tech. rep., Technical Report TUD-BS-1999-02, Darmstadt University of Technology (1999)

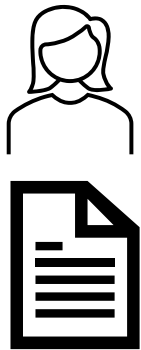
More specific exchange setting



Express $\Pi_{\text{doc}}()$ as a circuit



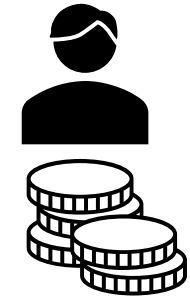
I'll trade my digital item for p coins

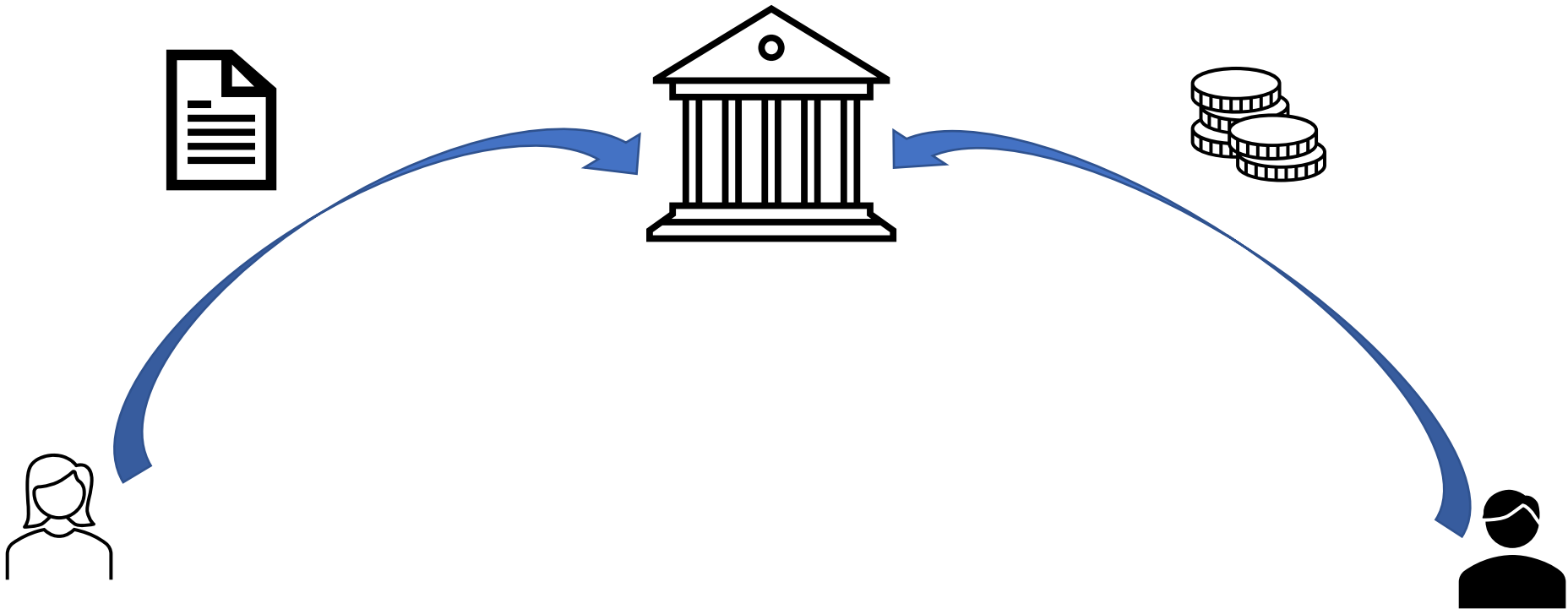


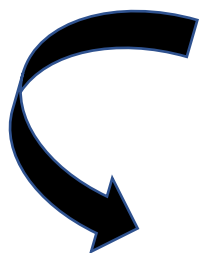
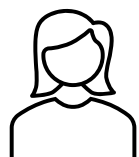
Divide doc into n chunks of Ω bits each for $n = 4$

$$\text{doc} = x = (x_1, x_2, x_3, x_4)$$

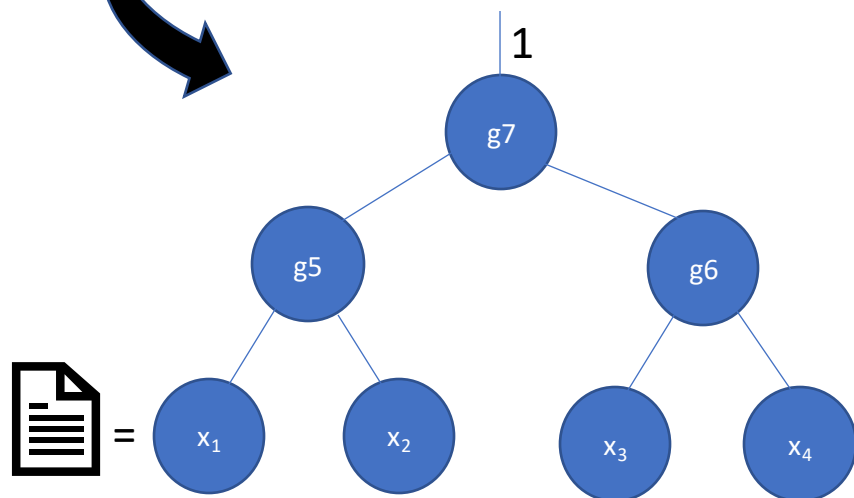
I'll trade p coins for a digital item that satisfies Boolean predicate $\Pi_{\text{doc}}()$

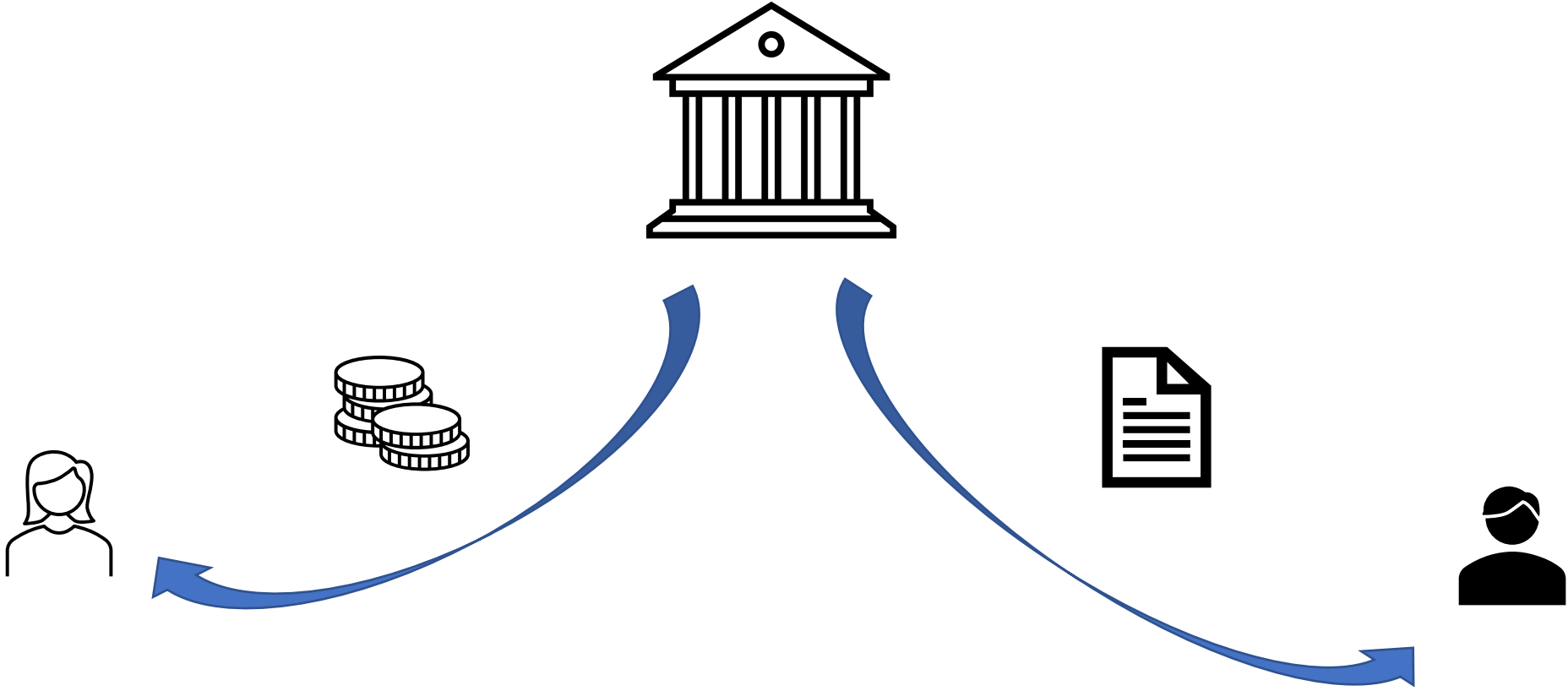






$$\pi_{\text{doc}}(\text{doc}) = 1 \quad \& \quad \text{coins} = p$$



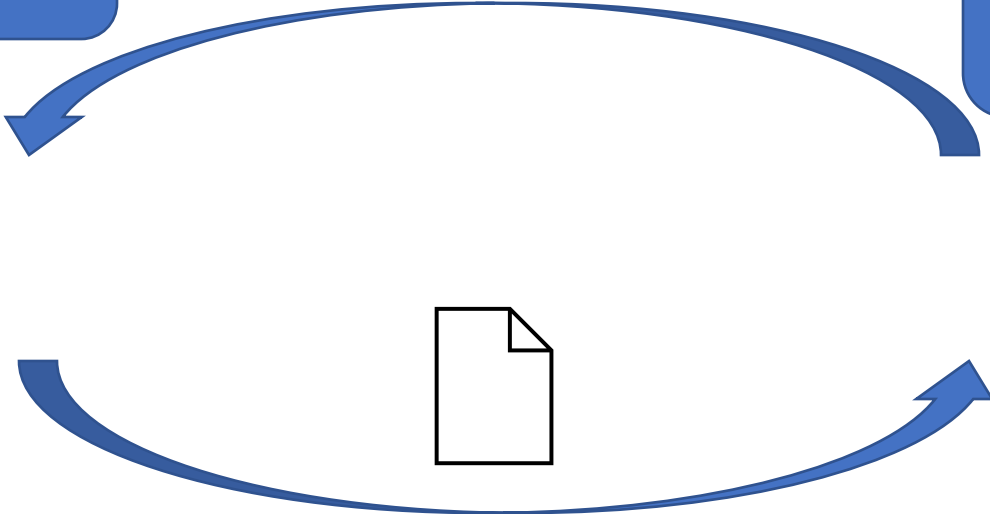
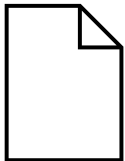


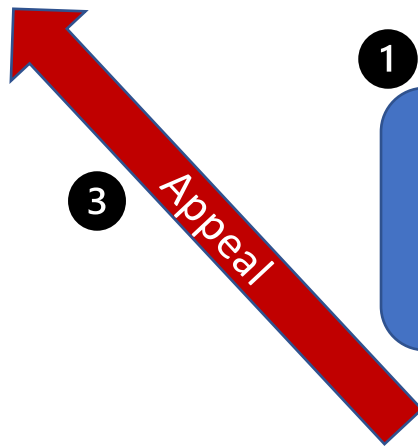
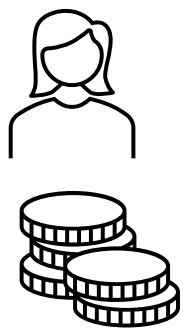
Optimistic Exchange



I'll trade my digital item for p coins

I'll trade p coins for a digital item that satisfies Boolean predicate $\Pi_{\text{doc}}()$

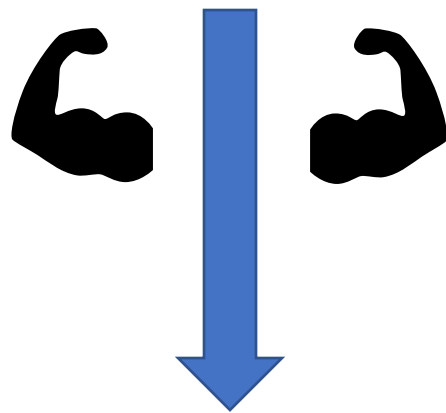




1
I'll trade p coins for a digital item that satisfies Boolean predicate $\prod_{\text{doc}}()$



2
 $\prod_{\text{doc}}(\text{doc}) = 0$

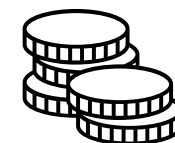


Either



Such that $\prod_{\text{doc}} (\text{doc}) = 1$

Or





REVIEWS ∨

NEWS ∨

TECH ∨

MONEY ∨

WELLNESS ∨

HOME ∨

CARS ∨

DEALS ∨

Teen pays \$735 for photo of Xbox One on eBay

A British teenager gets suckered out of \$735 when attempting to buy a Day One special-edition Xbox One console on eBay.

<https://www.cnet.com/news/teen-pays-735-for-photo-of-xbox-one-on-ebay/>

(Accessed Feb 10th 2022, Story from 2013)

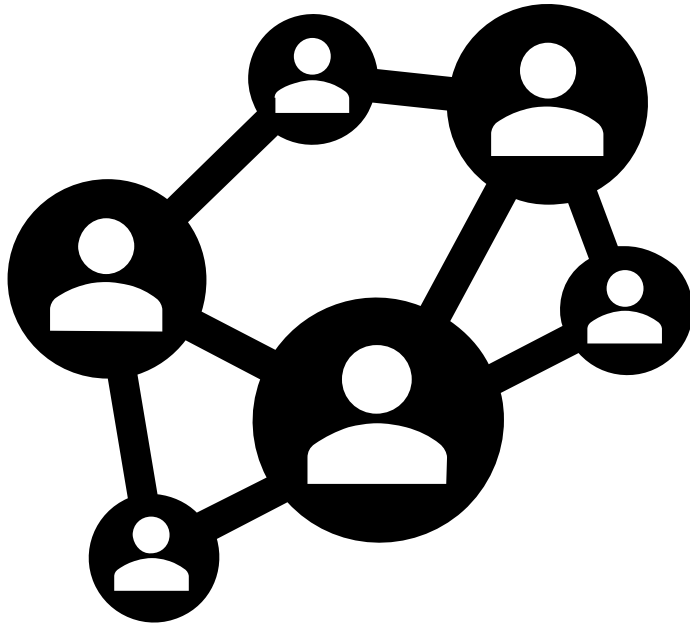
Replacing TTP with Blockchains (Decentralized Trust)

Dziembowski, S., Ekey, L., Faust, S.: Fairswap: How to fairly exchange digital goods. In: Proceedings of the 2018ACMSIGSAC Conference on Computer and Communications Security. pp. 967–984. ACM(2018)

Ekey, L., Faust, S., Schlosser, B.: Optiswap: Fast optimistic fair exchange. In: Proceedings of the 15th ACM Asia Conference on Computer and Communications Security. pp.543–557 (2020)

What are blockchains?

Network of Distributed Nodes that maintain a ledger



Entry	From	To	Amt
1	Bob	Alice	7.00
2	Alice	Bob	2.00
3	Charlie	Alice	6.00
4	Bob	Charlie	10.00





Entry	From	To	Amt
1	Bob	Alice	67.00
2	Alice	Bob	4.00
3	Charlie	Alice	9.00
4	Bob	Charlie	12.00



Entry	From	To	Amt
1	Bob	Alice	41.00
2	Alice	Bob	25.00
3	Charlie	Alice	12.00
4	Bob	Charlie	18.00

Accounts on blockchains use public / private key cryptography.

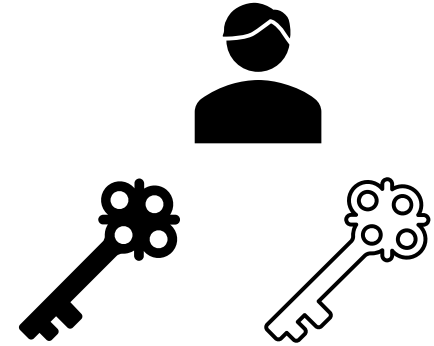
Digital signatures:

Generate a signature sig on a message M using a **private key** . The signature can be verified with the **public key** .

$(\text{private key}, \text{public key}) \leftarrow \text{KeyGen}(1^k)$

$sig \leftarrow \text{Sign}(M, \text{private key})$

$\text{Verify}(M, sig, \text{public key}) := 1/0$

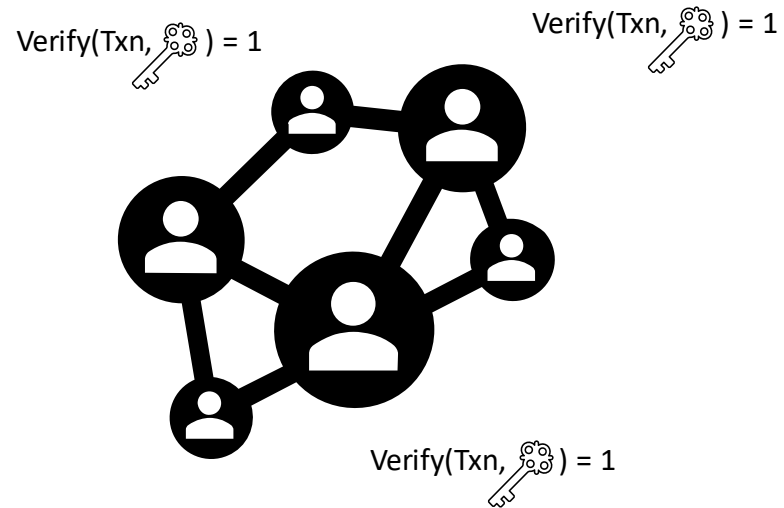
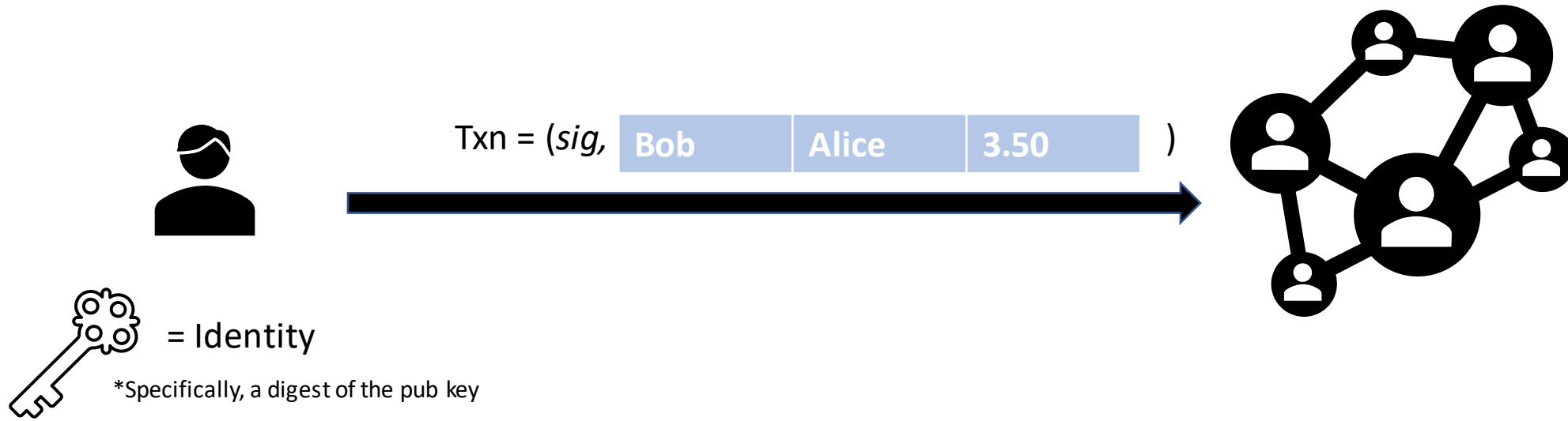


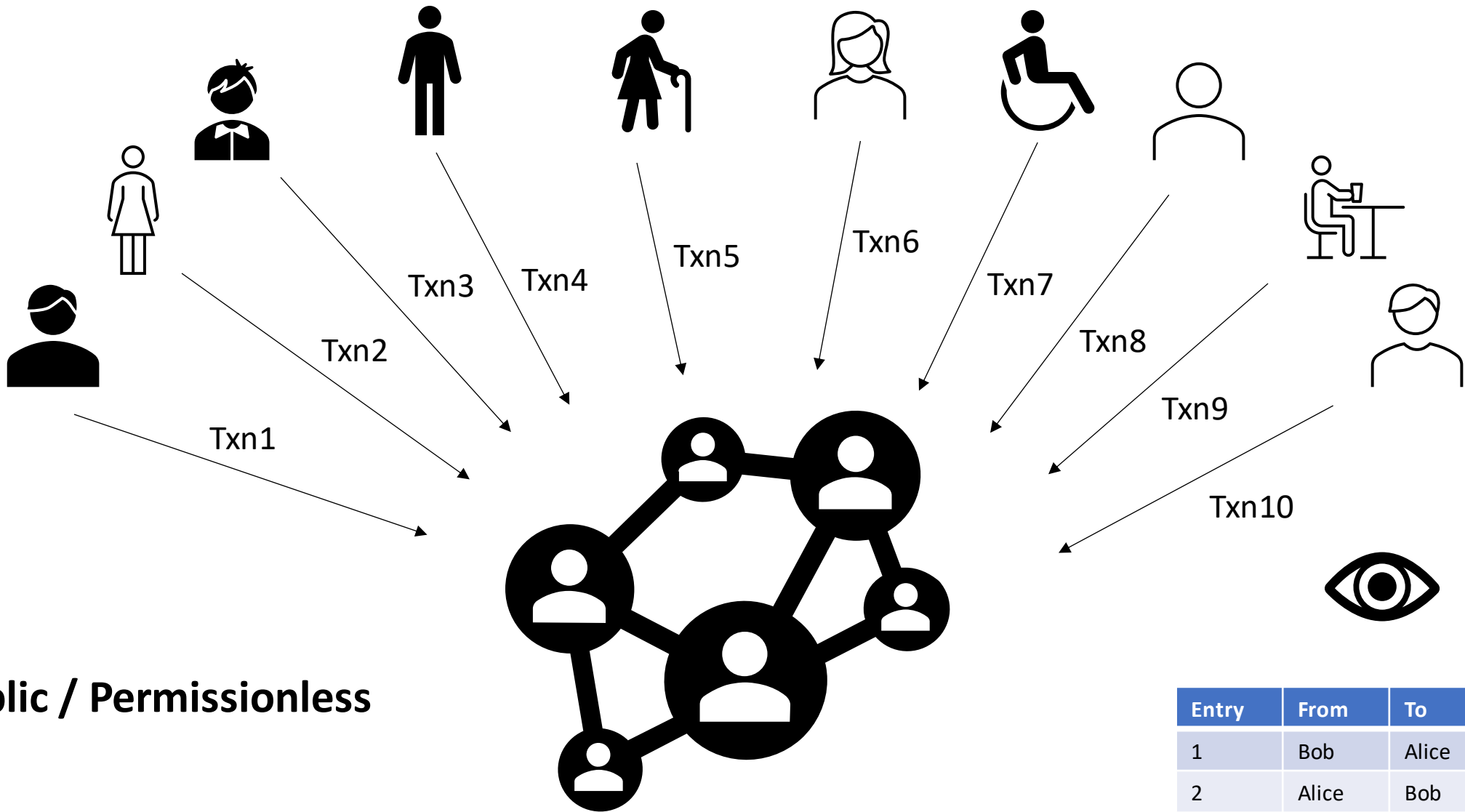
$M =$

Bob	Alice	3.50
-----	-------	------

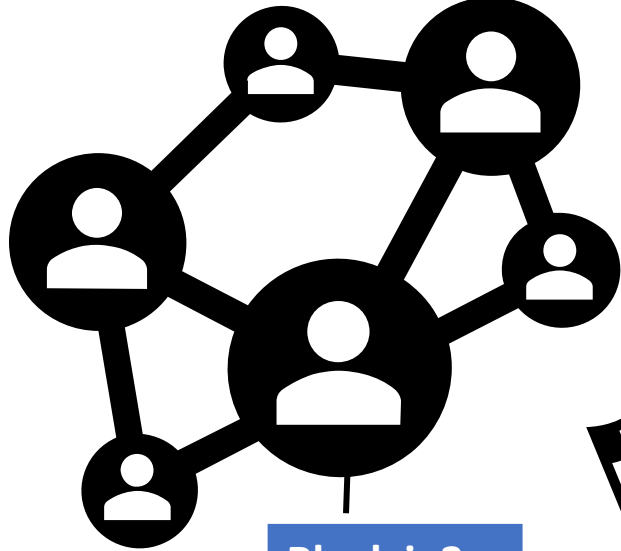
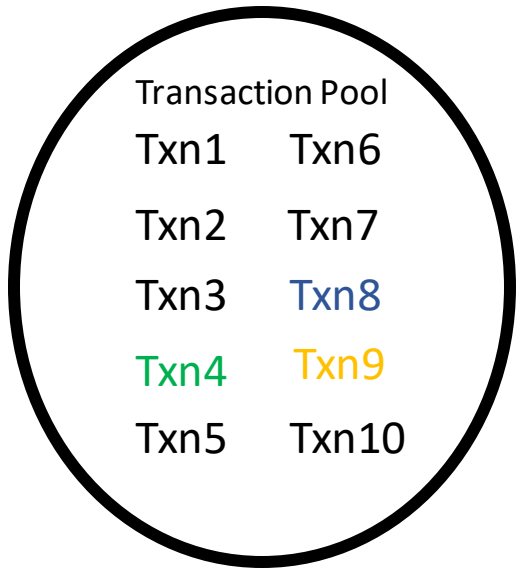
$sig = \text{Sign}(\text{Bob Alice 3.50 } \text{private key})$

What are blockchains?

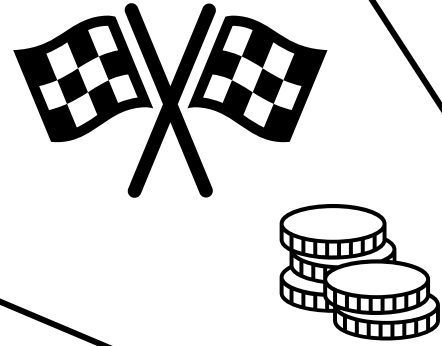




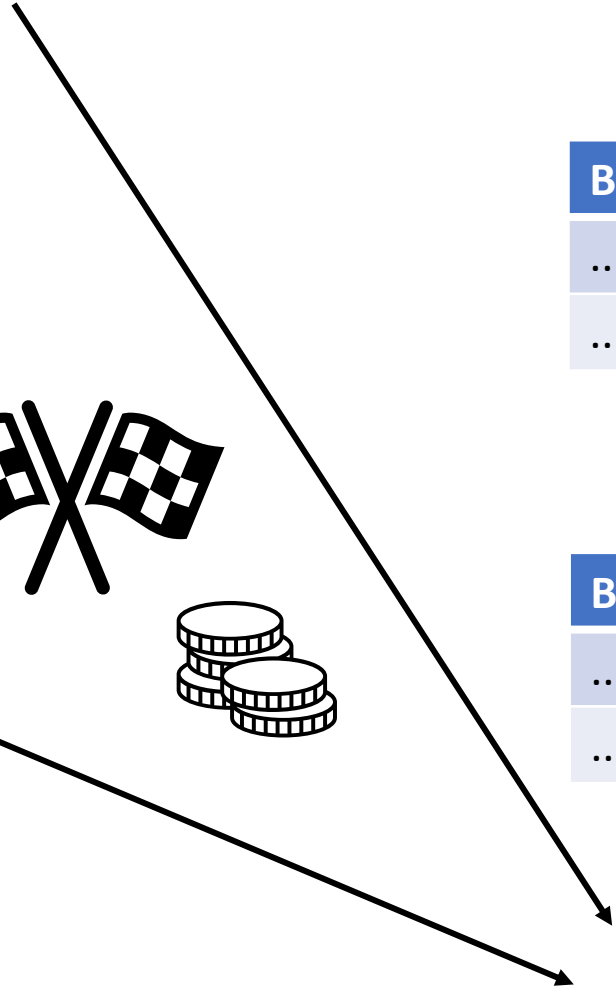
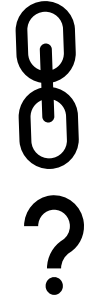
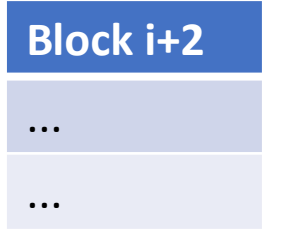
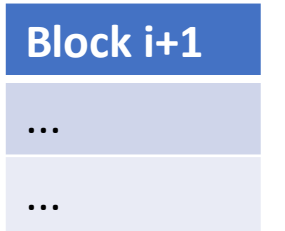
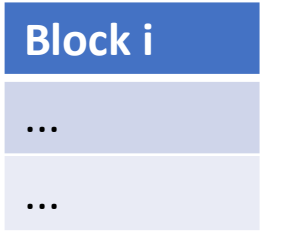
Entry	From	To	Amt
1	Bob	Alice	7.00
2	Alice	Bob	2.00
3	Charlie	Alice	6.00
4	Bob	Charlie	10.00



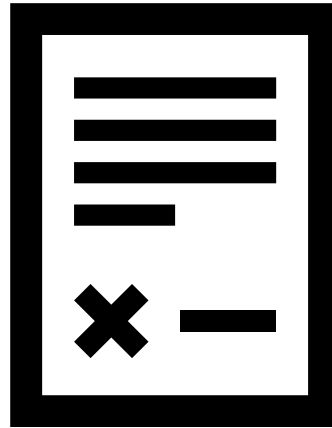
Txn set {



Hash(Txn set || nonce) = 00000...A4DF9BC0



Using smart contracts run on blockchains

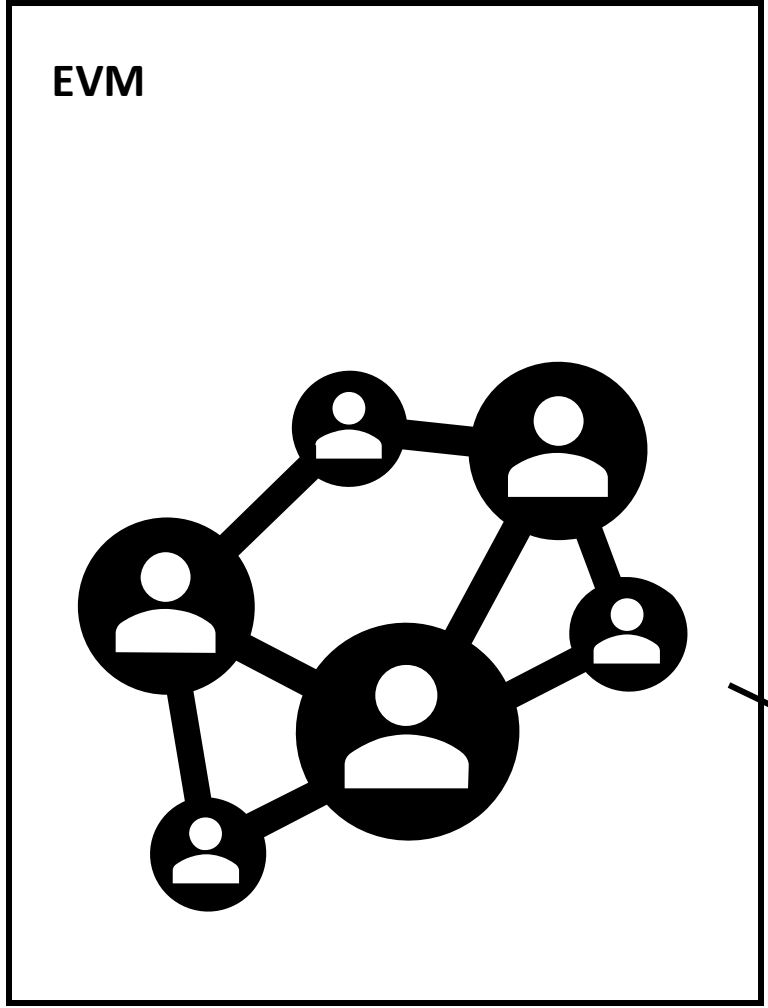
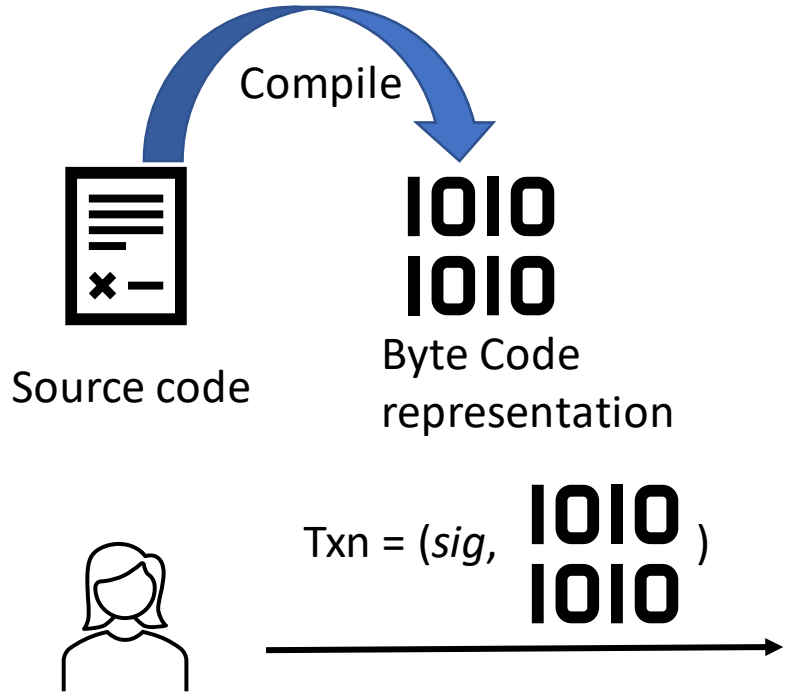


```
contract TTP {  
  global item_price = 200;  
  ...  
  function appeal(complaint)  
    IF complaint == True:  
      resolve_dispute();  
}
```

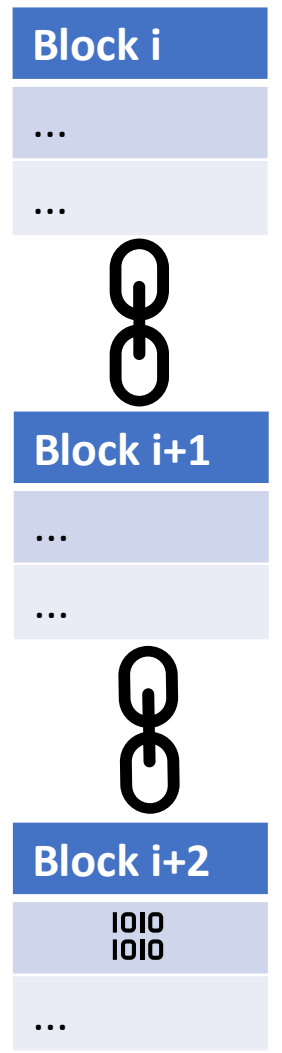
Binding Terms and Conditions



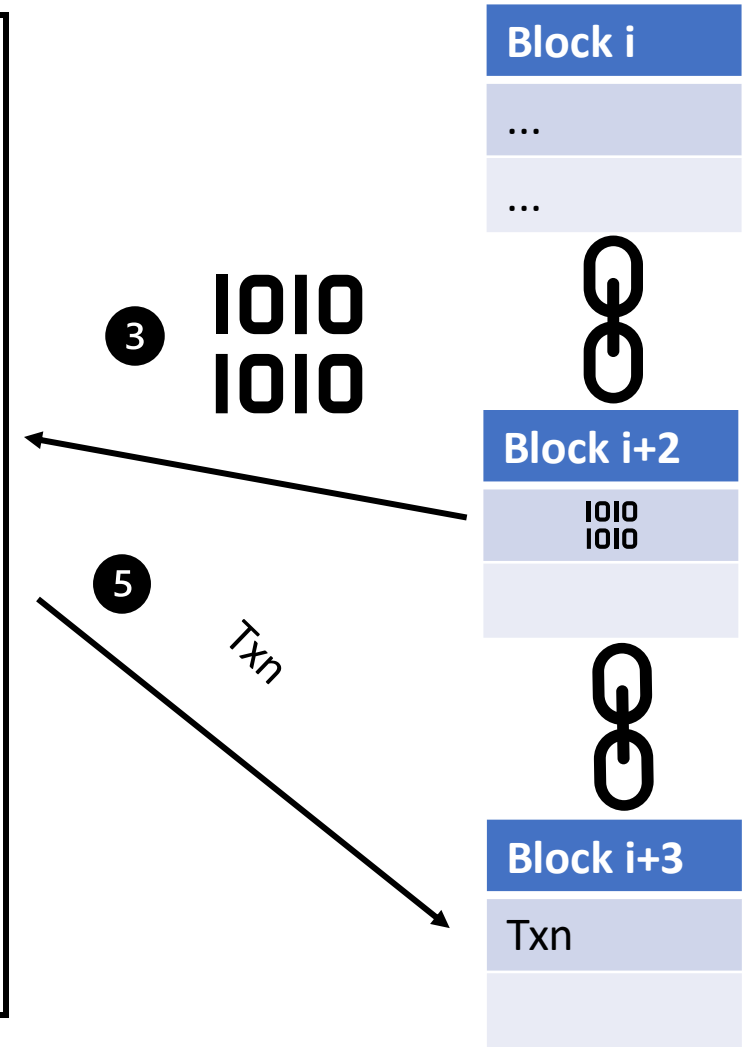
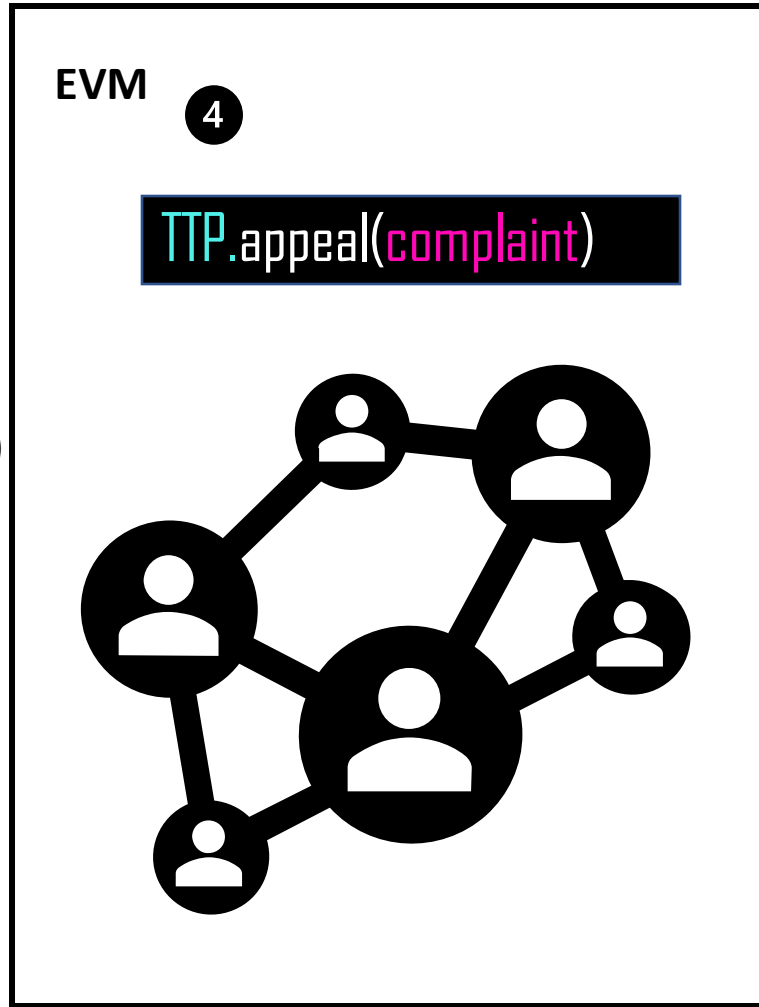
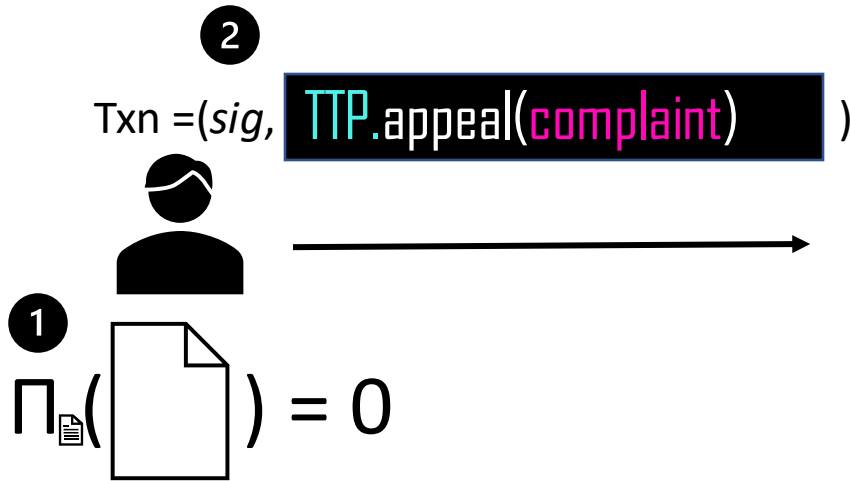
Smart Contracts



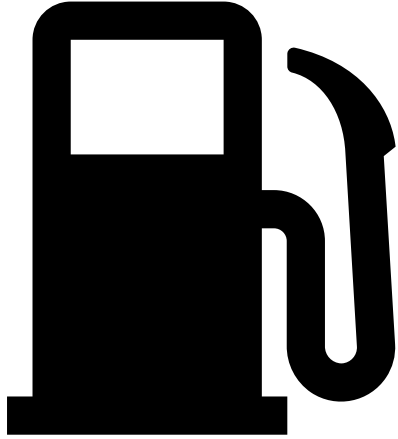
1010
1010



Ethereum Virtual Machine (EVM)
Global state machine

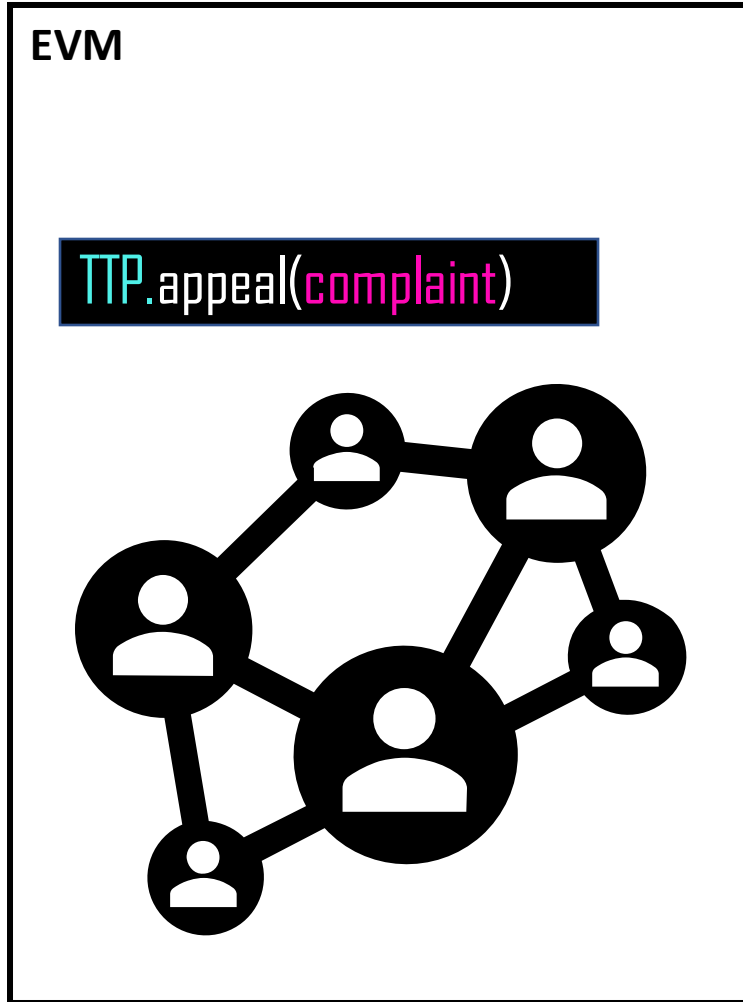


Still Public / Permissionless



Computing and Storage
"on chain" in the EVM is
expensive and requires
Gas to be paid.

OpCode	Gas
ADD	3
SUB	3
LOAD	4



Hash function

$H()$

- Arbitrarily large input
- Fixed size output
- Preimage resistance
- Second preimage resistance
- Collision resistance

Commit to data x with $H()$

$$H(x) = h$$

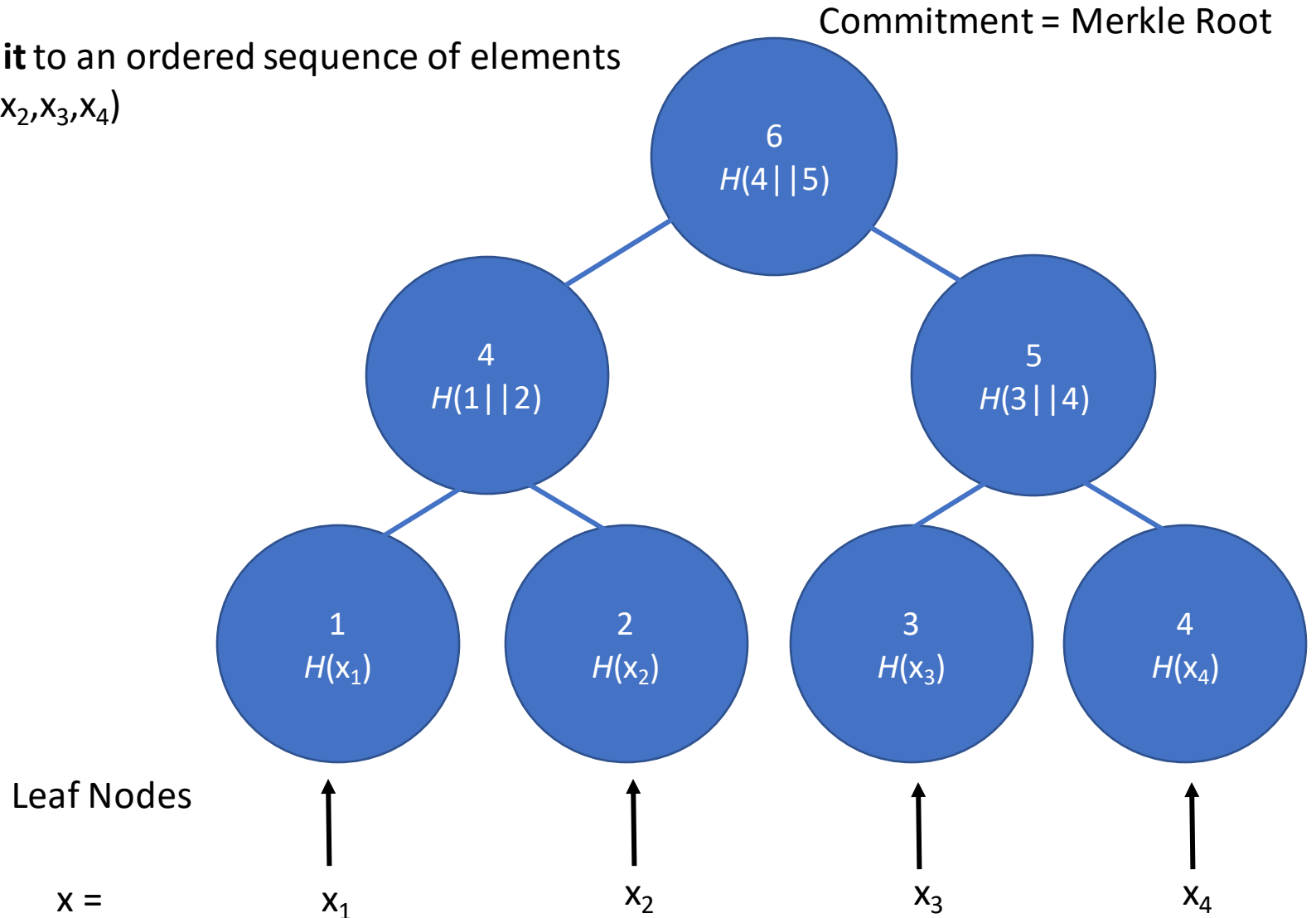
For x' where $H(x') = h'$

If $h' = h$, very likely $x = x'$

Merkle Tree

Commit to an ordered sequence of elements

$$x = (x_1, x_2, x_3, x_4)$$

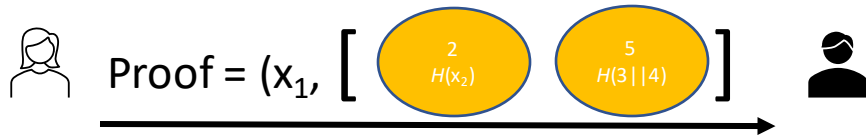


Use Merkle Trees to prove that an element x_i belongs to the sequence at position i

First: Alice commits to sequence x with Merkle root



Later: Alice proves x_1 belongs to x at position 1



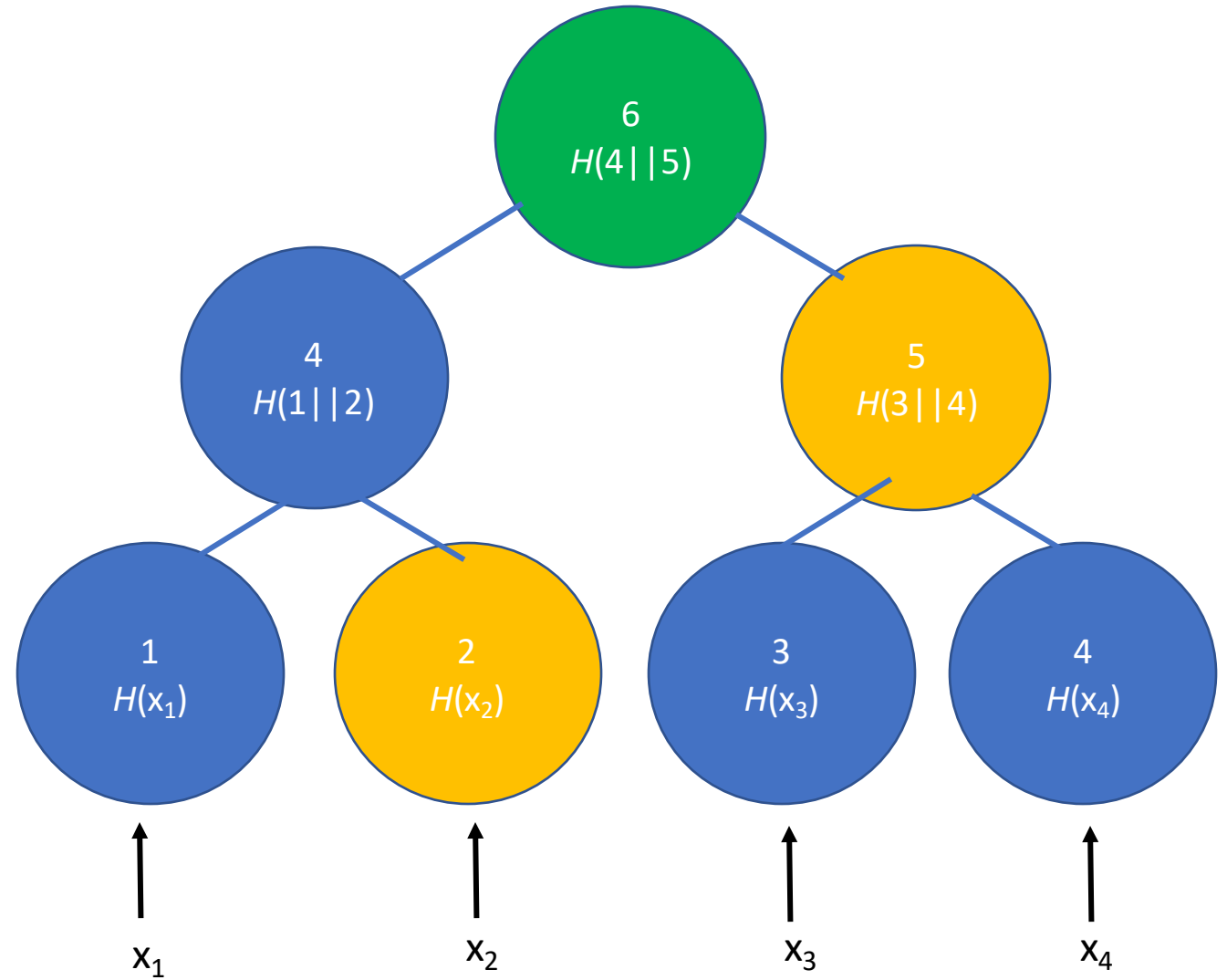
Verify(Proof):

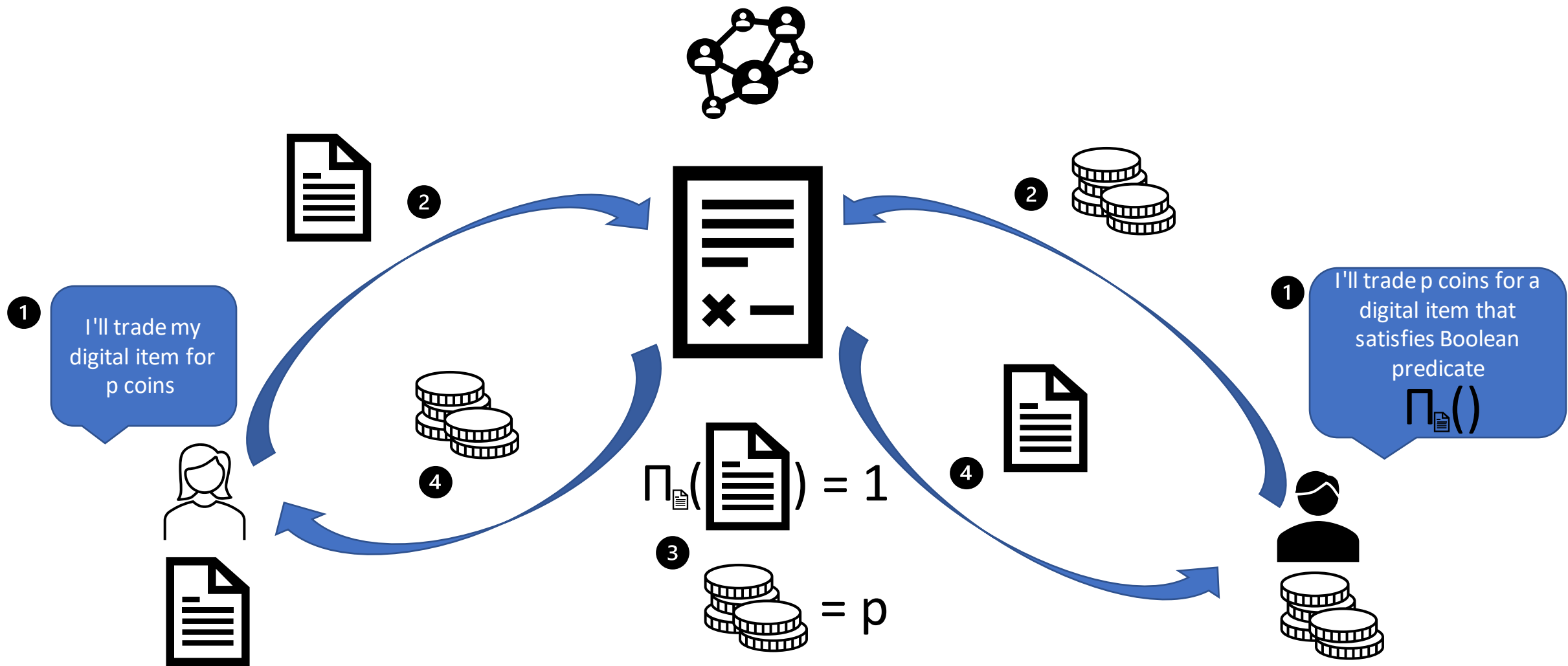
$$y_1 = H(x_1)$$

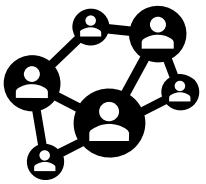
$$y_2 = H(y_1 || H(x_2))$$

$$\text{return } H(y_2 || H(3 || 4)) == H(4 || 5)$$

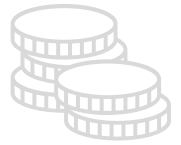
Merkle Tree



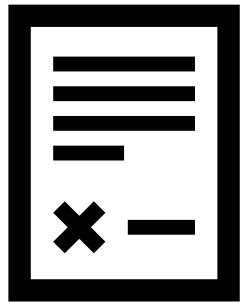




2



2



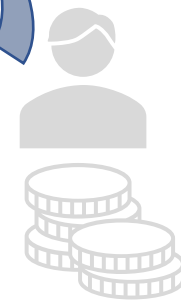
1

I'll trade my digital item for p coins



1

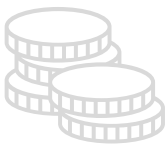
I'll trade p coins for a digital item that satisfies Boolean predicate $\Pi_{\text{doc}}()$



4

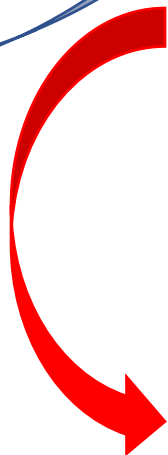
$$\Pi_{\text{doc}}(\text{doc}) = 1$$

3



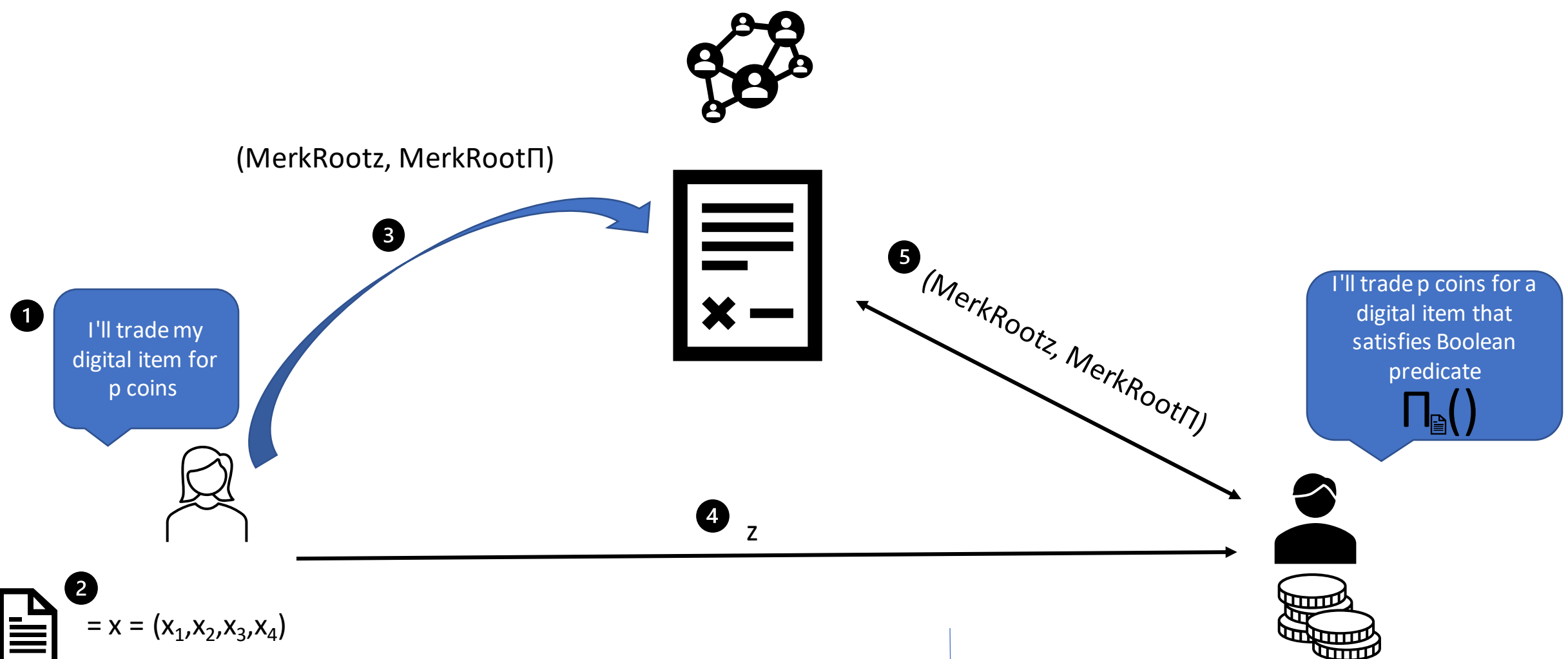
= p

4



If $\Pi_{\text{doc}}()$ or doc are large, Gas to verify item is prohibitively expensive!!





2 = $x = (x_1, x_2, x_3, x_4)$

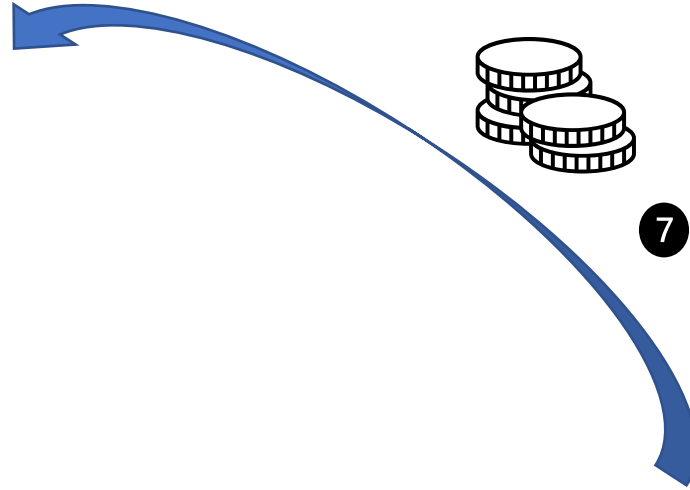
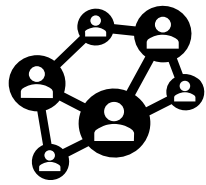
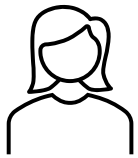
$\Pi_{\text{doc}}() = [in_1, in_2, in_3, in_4, g_5, g_2, g_3]$

$\Pi_{\text{doc}}(\text{doc})_{\text{exe}} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]$

$\text{Enc}(\text{exe}, \text{key}) = z = (z_1, z_2, z_3, z_4, z_5, z_6, z_7)$

MerkCommit(z) = MerkRootz
 MerkCommit($\Pi_{\text{doc}}()$) = MerkRoot Π

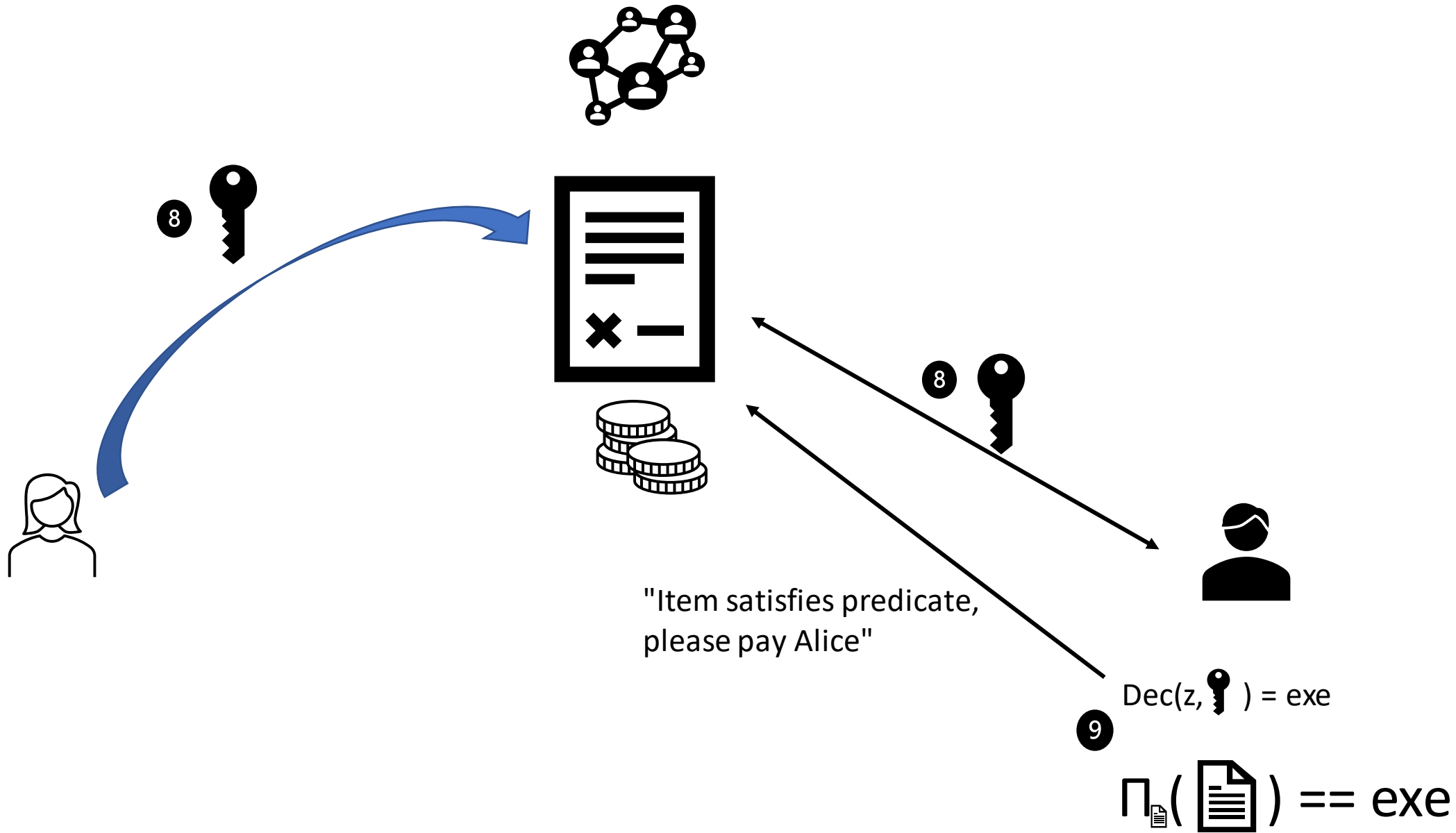
6 Check:
 MerkCommit($\Pi_{\text{doc}}()$) == MerkRoot Π
 MerkCommit(z) == MerkRootz

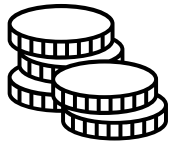
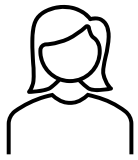


IF checks pass:

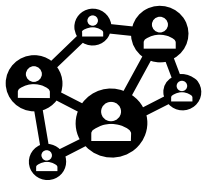
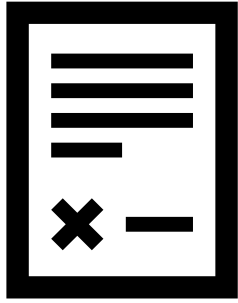
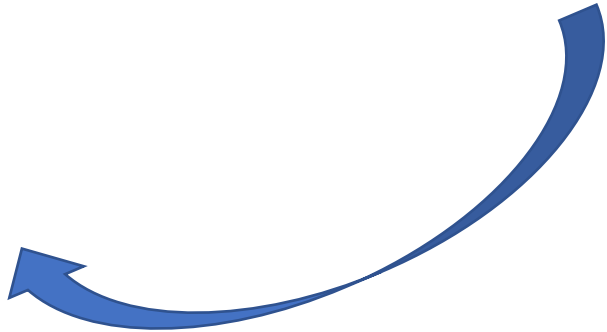
$$\text{MerkCommit}(\prod_{\text{document}}()) == \text{MerkRoot}\Pi$$

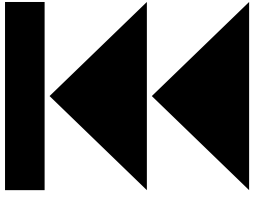
$$\text{MerkCommit}(z) == \text{MerkRoot}z$$





10





Rewind: What happens when

9

$\cap(\text{document}) \neq \text{exe}$

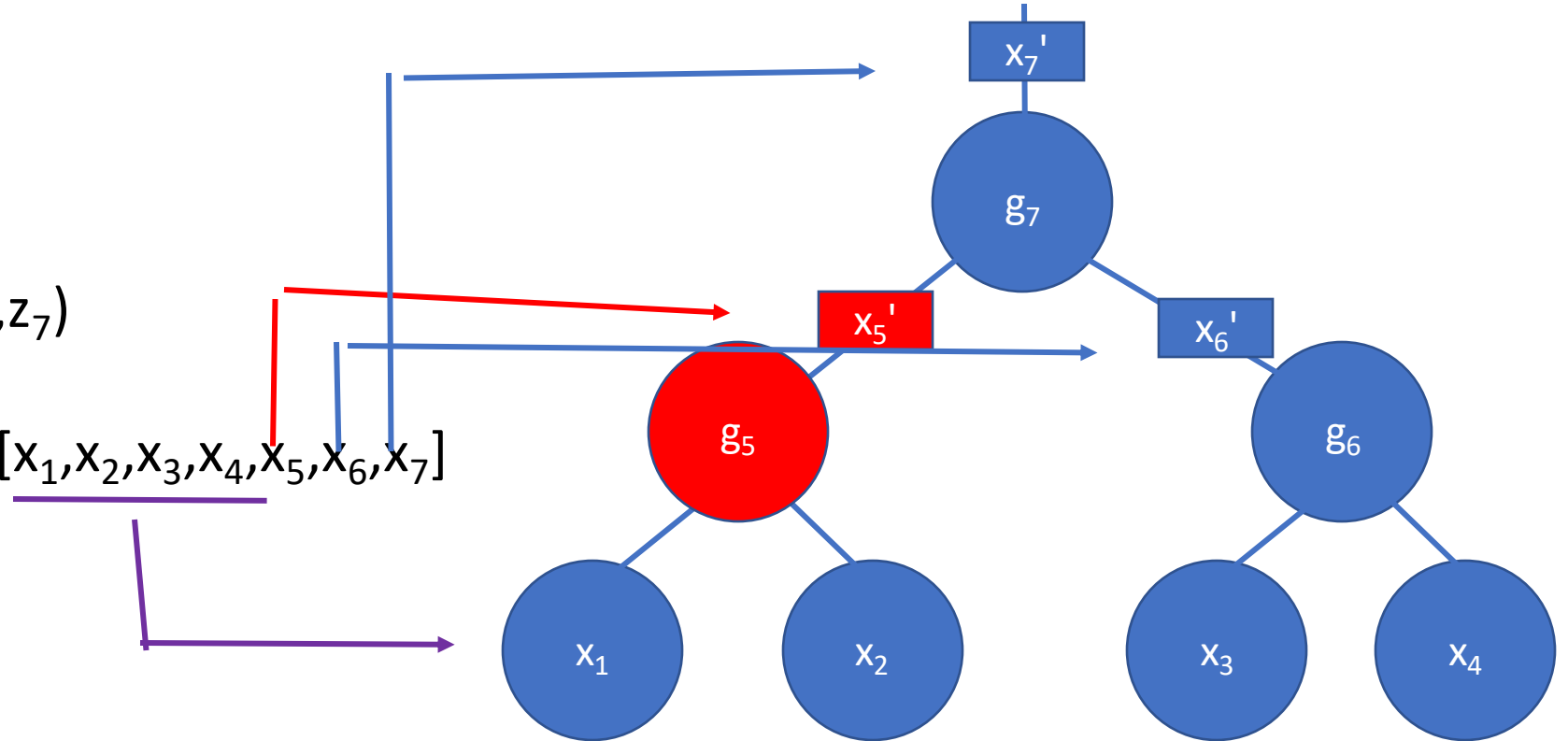
Recall:

$z = (z_1, z_2, z_3, z_4, z_5, z_6, z_7)$

$\text{Dec}(z, \text{key}) = \text{exe} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]$



$= x = (x_1, x_2, x_3, x_4)$



Recall:

$z = (z_1, z_2, z_3, z_4, z_5, z_6, z_7)$ (MerkRootz)

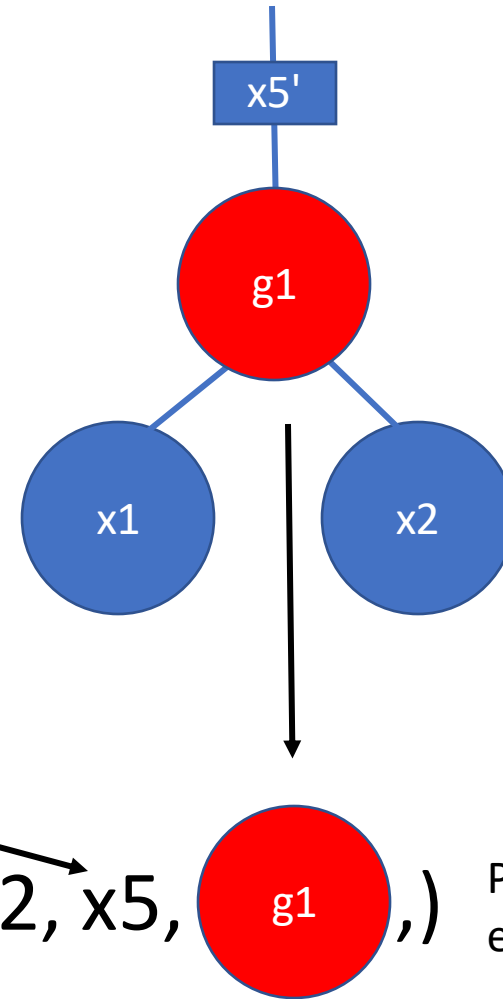
$\Pi_{\text{doc}}() = [\text{in}_1, \text{in}_2, \text{in}_3, \text{in}_4, g_1, g_2, g_3]$ (MerkRoot Π)

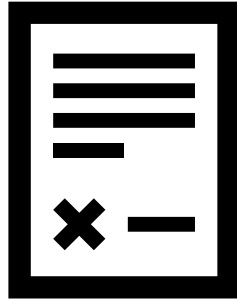
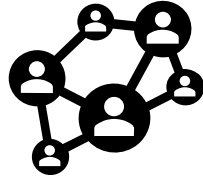
$\text{Dec}(z, \text{key}) = \text{exe} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]$



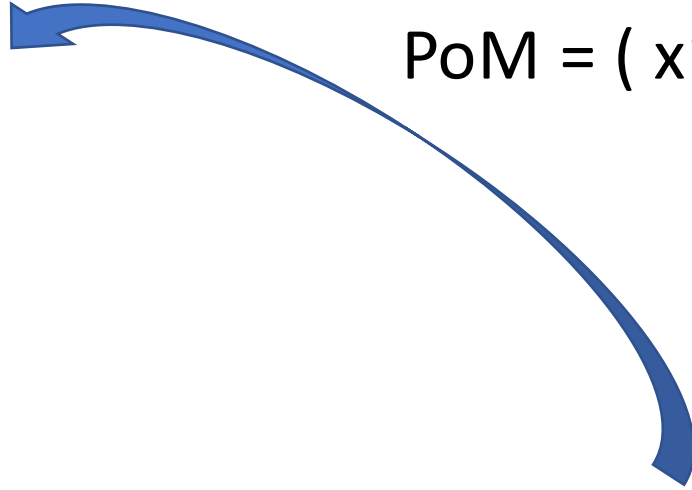
Proof of Misbehavior = $(x_1, x_2, x_5, g_1,)$

Plus Merkle Proofs for each element



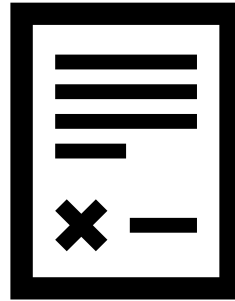
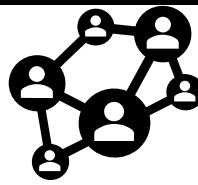


PoM = (x1, x2, x5, **g1**) + Merkle Proofs



Problem: Two inputs, and the output (3 wires) are made public.

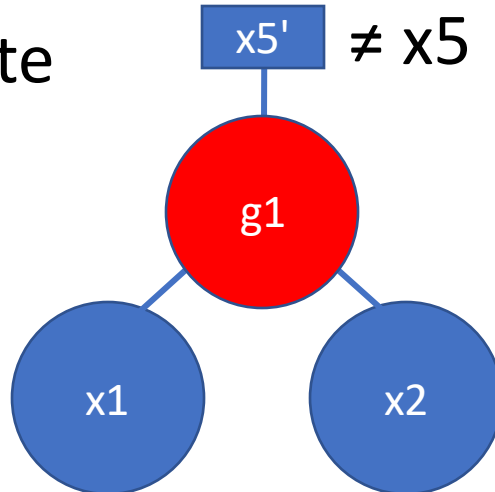
EVM



PoM = (x1, x2, x5, **g1**) + Merkle proofs

1) Verify all Merkle proofs

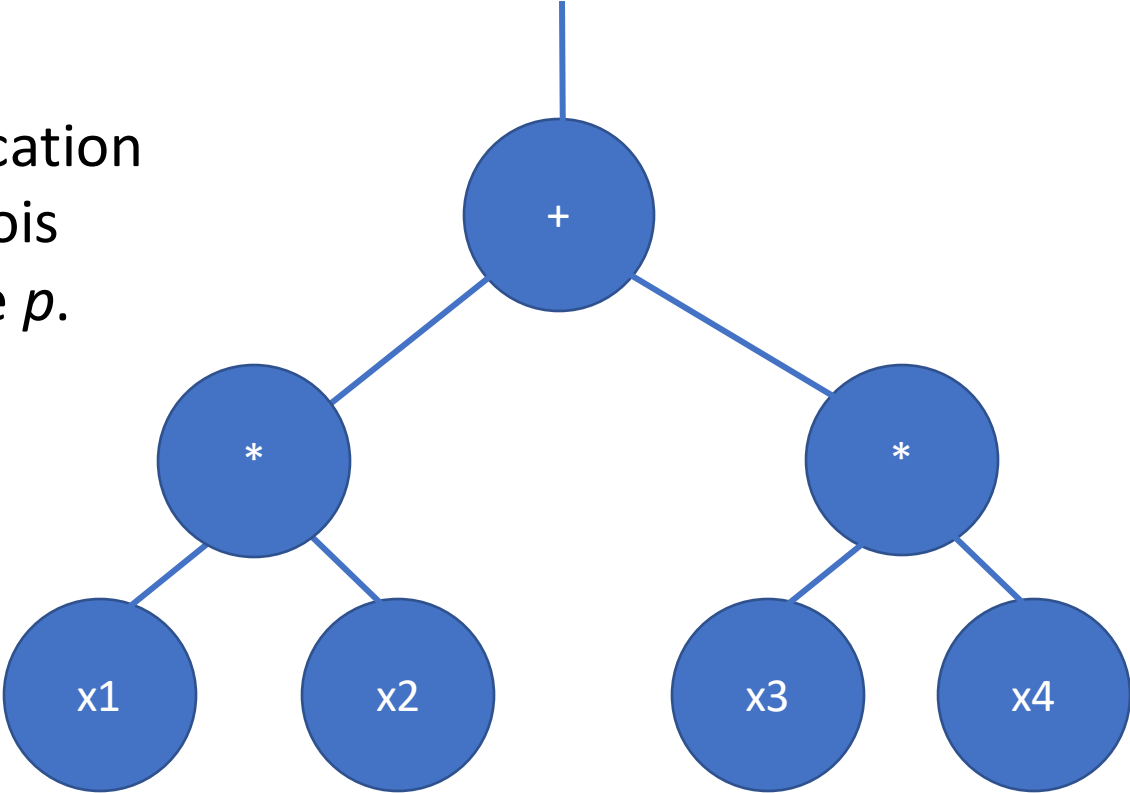
2) compute $x5' \neq x5$



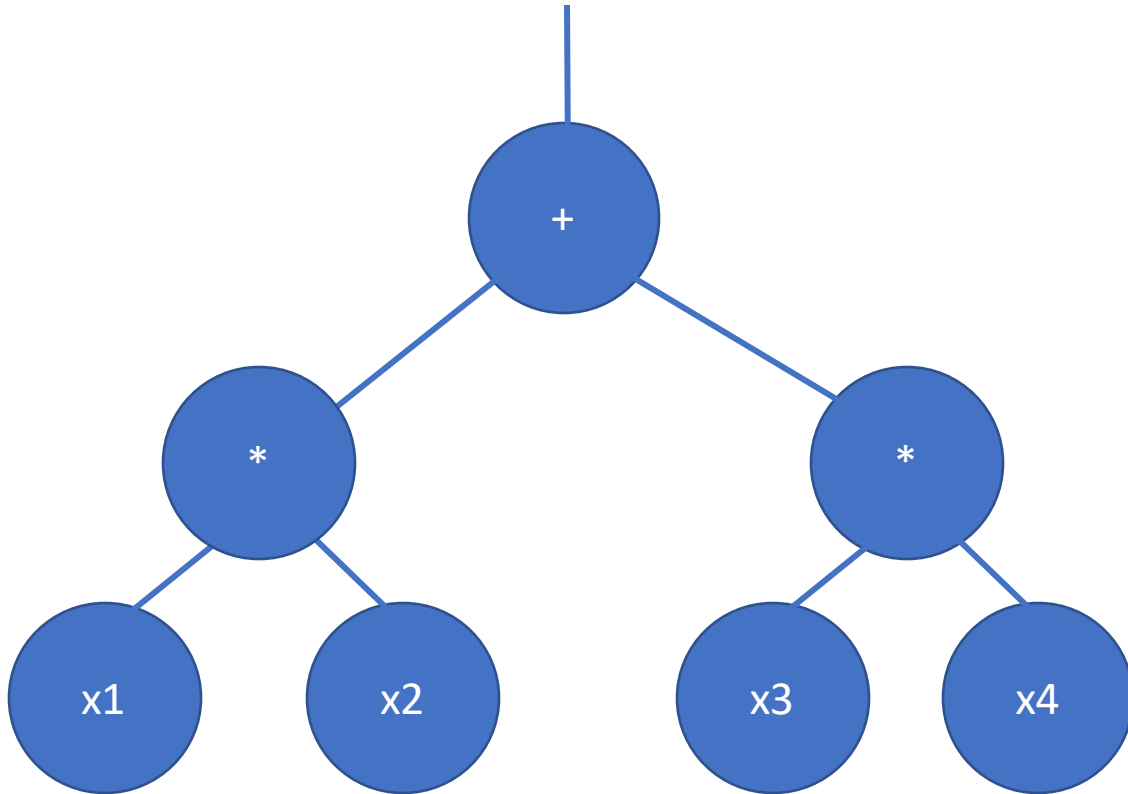
How can we show a single gate without revealing information about the computation?

Limit Our Selves to Arithmetic Circuits

Addition and Multiplication operations over a Galois Field $GF(p^n)$, for prime p .



Randomize the arithmetic circuit using $(m+1)$ -
 $(m+1)$ additive secret sharing



Input Encoding

Set $m = 2$

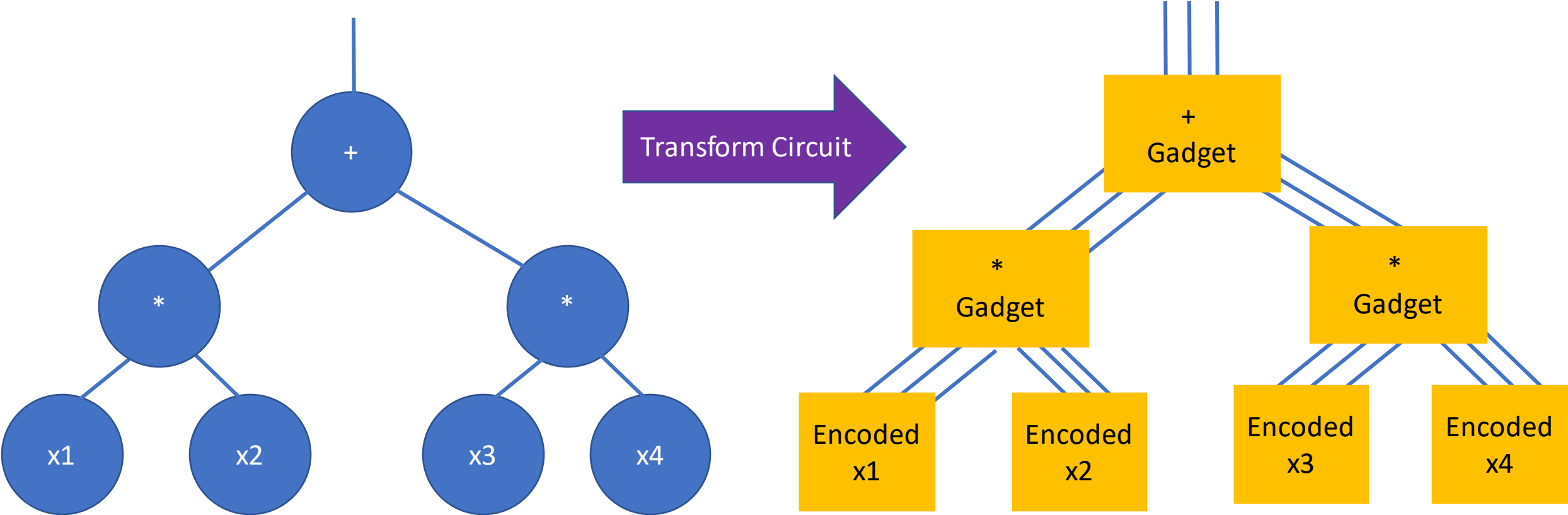
$x1 \rightarrow (r1, r2, r3)$

$$x1 = r1 + r2 + r3 \pmod{p^n}$$

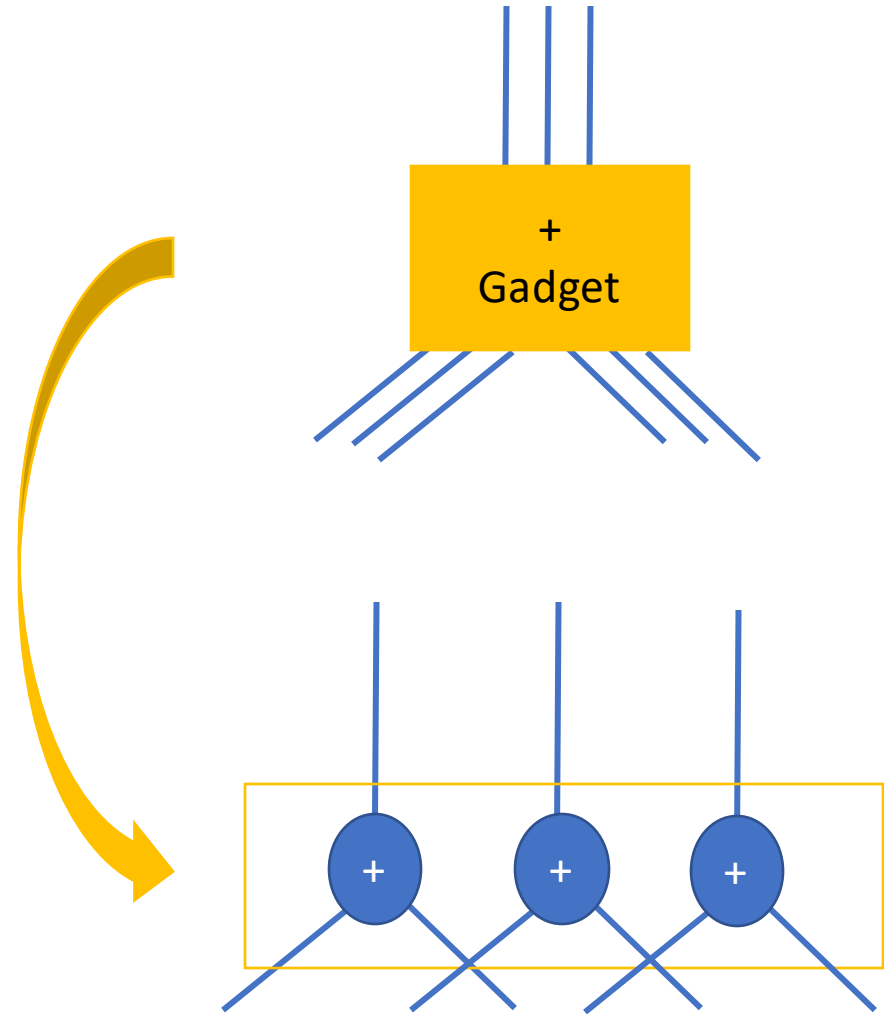
$$r3 = x1 - (r1 + r2)$$

$GF(p^n)$

We need a circuit that can execute on our inputs



$$\begin{array}{r} A = (a_1 + a_2 + a_3) \\ + B = (b_1 + b_2 + b_3) \\ \hline C = (c_1 + c_2 + c_3) \end{array}$$

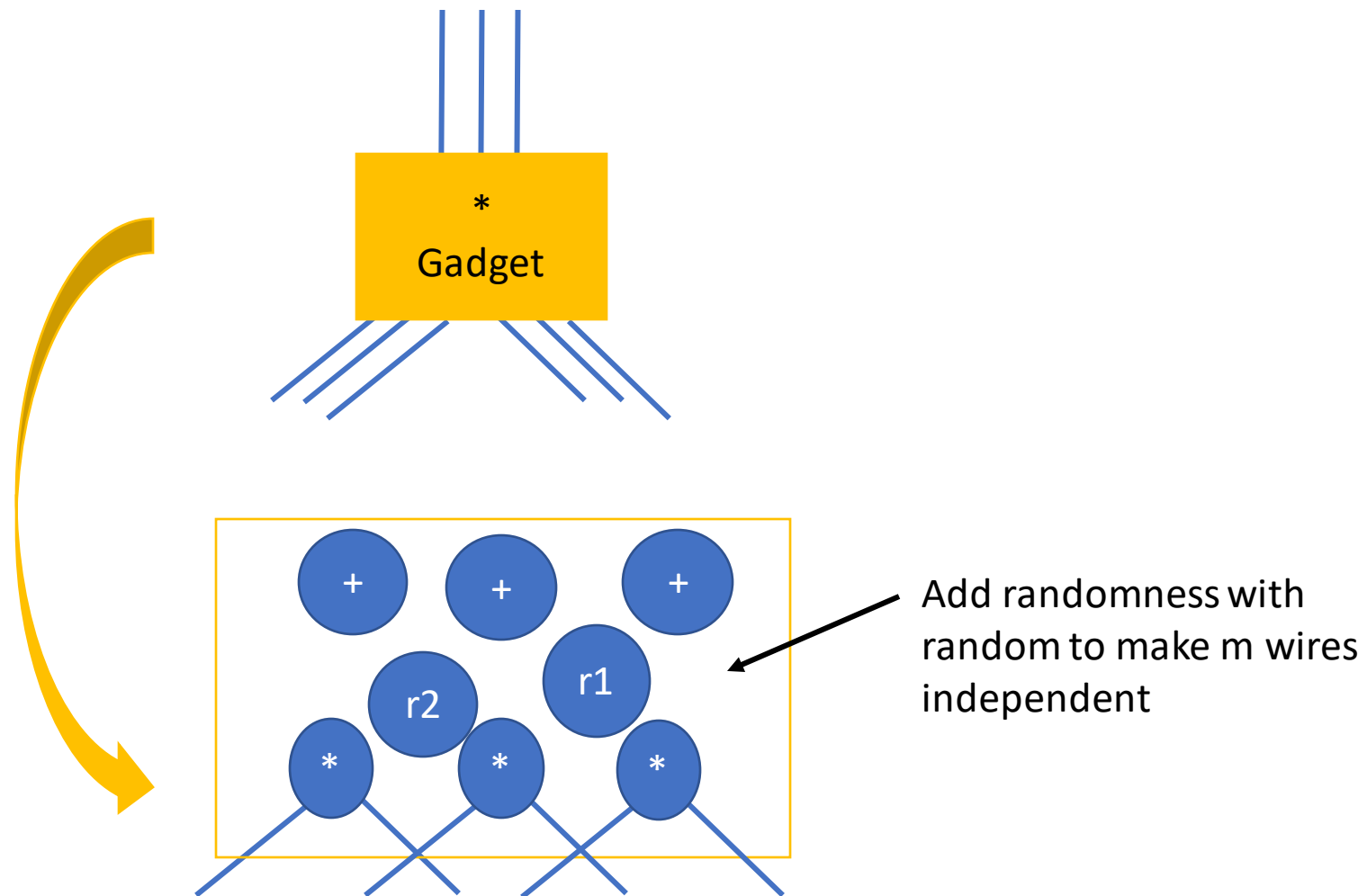


$$A = (a_1 + a_2 + a_3)$$

X

$$\ast B = (b_1 + b_2 + b_3)$$

$$C = (c_1 + c_2 + c_3)$$



More Info:

Avizheh, Sepideh & Haffey, Preston & Safavi-Naini, Reihaneh. (2022). Privacy-preserving FairSwap: Fairness and privacy interplay. *Proceedings on Privacy Enhancing Technologies*. 2022. 417-439. 10.2478/popets-2022-0021.

Sepideh Avizheh, Preston Haffey, and Reihaneh Safavi-Naini. 2021. Privacy-enhanced OptiSwap. *Proceedings of the 2021 on Cloud Computing Security Workshop*. Association for Computing Machinery, New York, NY, USA, 39–57. DOI:<https://doi.org/10.1145/3474123.3486756>