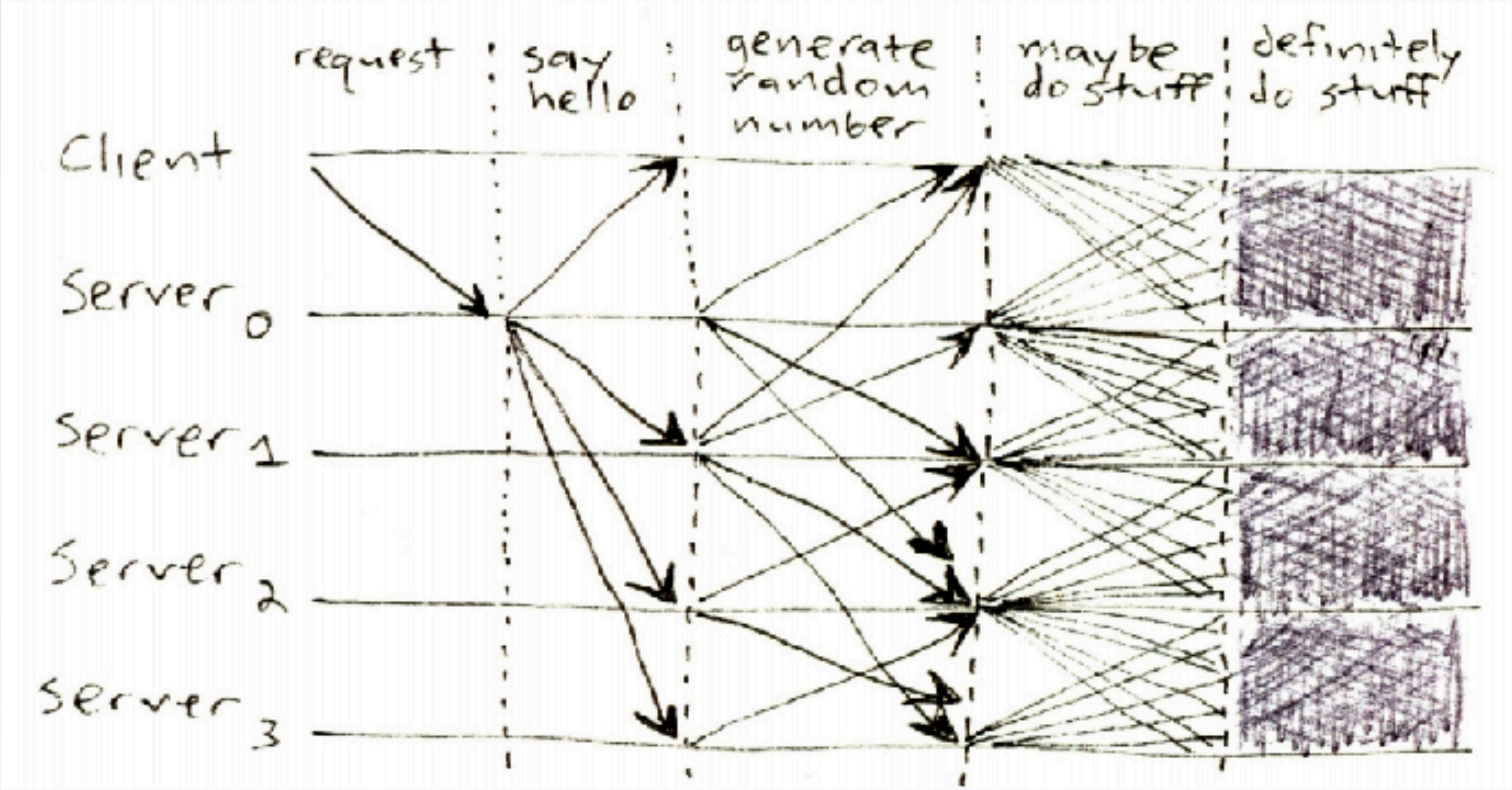
Blockchains & Cryptocurrencies

Towards Consensus



Instructors: Matthew Green & Abhishek Jain
Johns Hopkins University - Spring 2019

Some slides based on NBFMG

Housekeeping

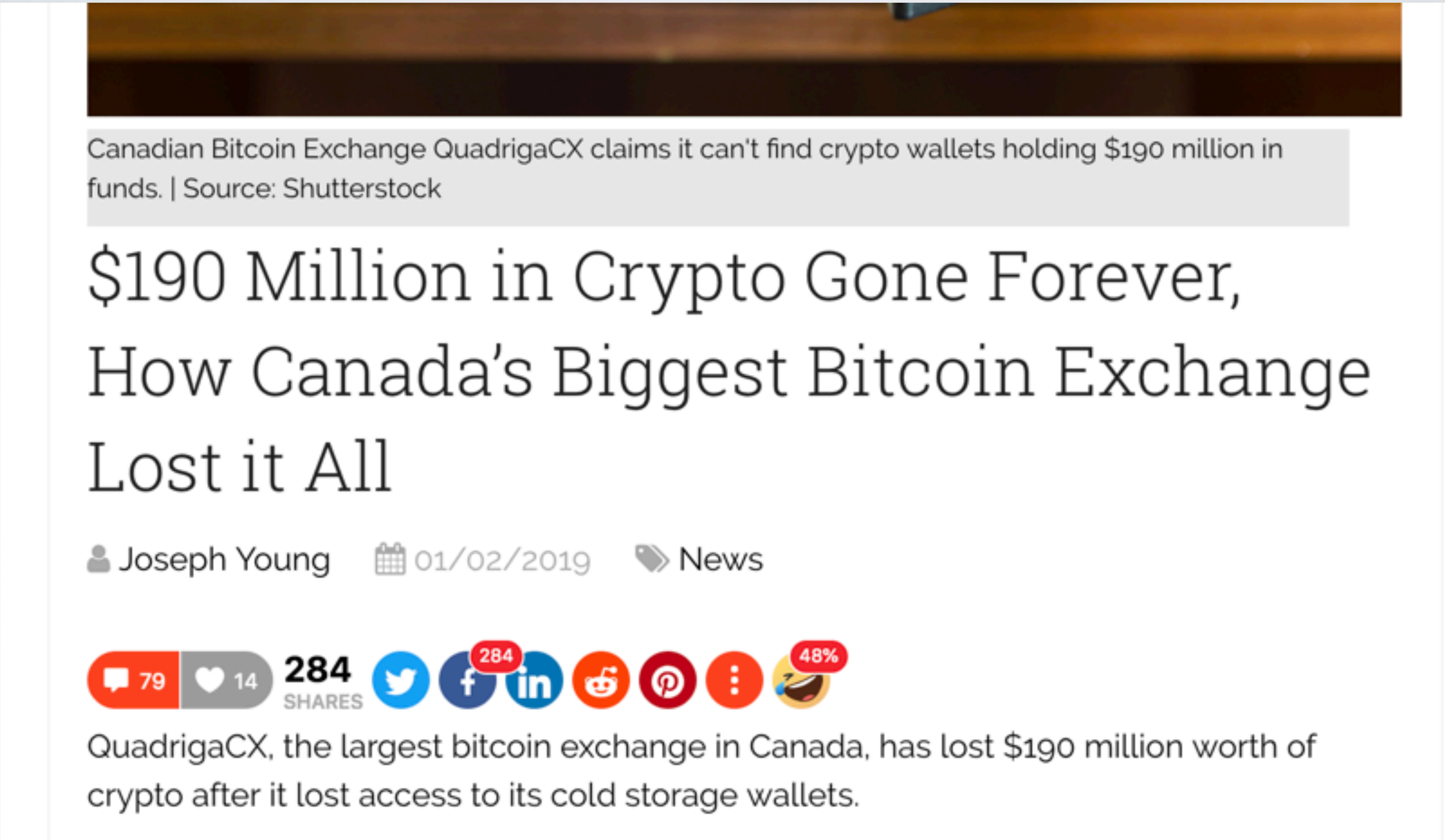
- **Assignment 1 is out** (last Weds)
- Due Friday 2/15 end of day
- Readings: for today crypto stuff, NBFMG 0,1
  for weds: Bitcoin and Hashcash
- Office hours, etc., Piazza, TA questions

News?

News?

- **Grin** launched 1/15
  - Privacy-preserving currency based on MimbleWimble
  - We'll talk about how this all works later

News?



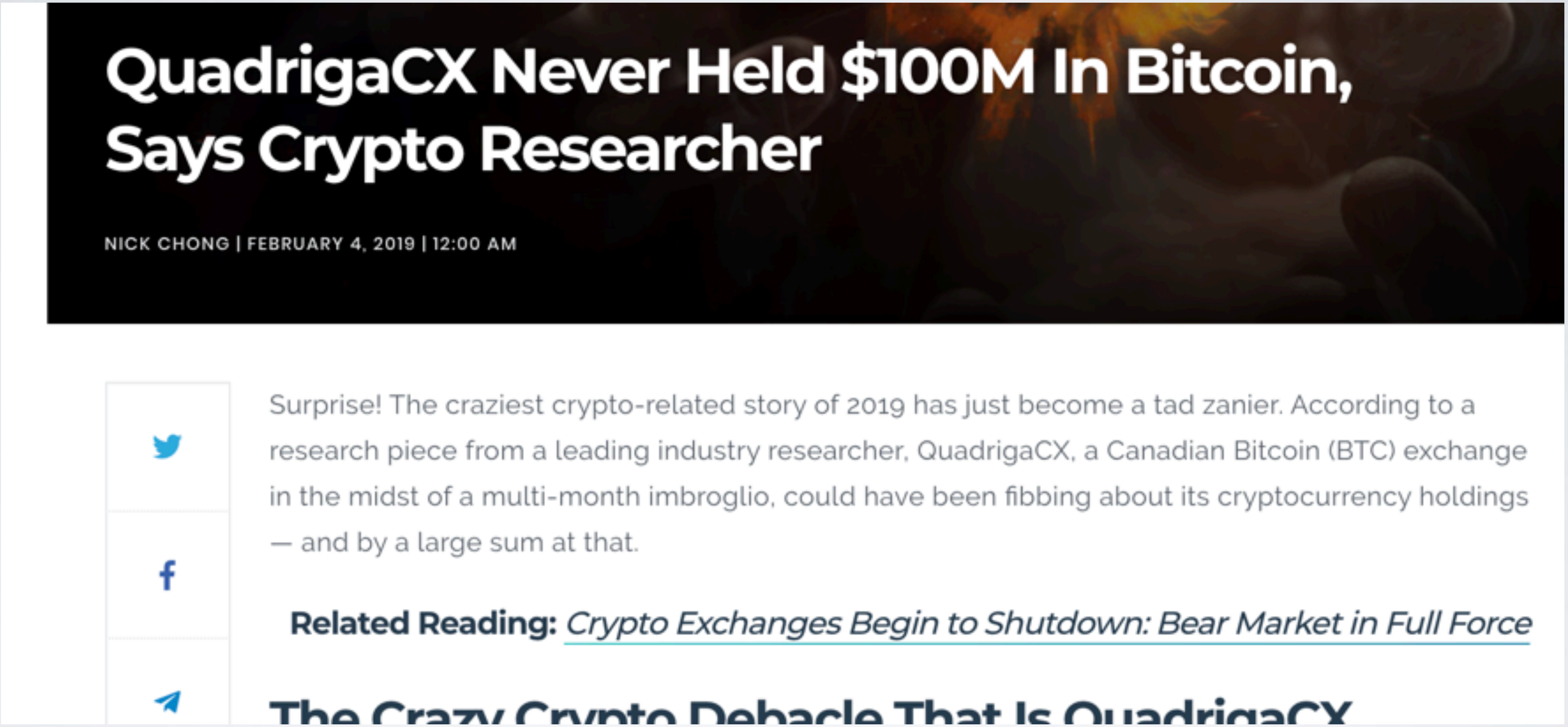
Canadian Bitcoin Exchange QuadrigaCX claims it can't find crypto wallets holding $190 million in funds. | Source: Shutterstock

$190 Million in Crypto Gone Forever, How Canada's Biggest Bitcoin Exchange Lost it All

Joseph Young 01/02/2019 News

QuadrigaCX, the largest bitcoin exchange in Canada, has lost $190 million worth of crypto after it lost access to its cold storage wallets.

News?



QuadrigaCX Never Held $100M In Bitcoin, Says Crypto Researcher

NICK CHONG | FEBRUARY 4, 2019 | 12:00 AM

Surprise! The craziest crypto-related story of 2019 has just become a tad zanier. According to a research piece from a leading industry researcher, QuadrigaCX, a Canadian Bitcoin (BTC) exchange in the midst of a multi-month imbroglio, could have been fibbing about its cryptocurrency holdings — and by a large sum at that.

Related Reading: *Crypto Exchanges Begin to Shutdown: Bear Market in Full Force*

The Crazy Crypto Debacle That Is QuadrigaCX

https://medium.com/@zeroresearchproof/quadrigacx-chain-analysis-report-pt-1-bitcoin-wallets-19d3a375d389

Today

- We're going to talk about "consensus"
- What the hell is consensus, how do you accomplish it, what's the point?
- This is all in preparation for Weds, when we'll actually talk about Bitcoin

Review: cash problems

- **Double spending**
  - To capture double spending you need an online (networked) party that must be trusted
- **Privacy**
  - In many naive systems, the bank sees every transaction you make
- **Origin/Issuance**
  - How is new currency created?

Review: centralized $$

- Use a centralized bank database ("ledger") to record account balances
  - Require merchants/ATMs to contact the bank for approval
  - Ledger can be "account-based" or "transaction-based"
    - Typically it's both, and the two are reconciled

Soaring Eagle Casino and
6800 Soaring Eagle Blvd MT. PLEASANT MI 48858
CASHOUT TICKET
VALIDATION 00-8368-6231-8062-1546
10/13/2014 20:46:16 TICKET # 0237
TWO HUNDRED FIFTY TWO DOLLARS AND TWENTY FIVE CENTS
$252.25
Ticket Void after 90 days ASSET# 23208

Problems with centralized $$

- Centralized party is a huge trust requirement
  - This bank can attack the system and steal your money
  - The bank can be taken down by a DDoS or hacked, shut down by legal action
  - Also: who issues the currency?

Soaring Eagle Casino and
6800 Soaring Eagle Blvd MT. PLEASANT MI 48858
CASHOUT TICKET
VALIDATION 00-8368-6231-8062-1546
10/13/2014 20:46:16 TICKET # 0237
TWO HUNDRED FIFTY TWO DOLLARS AND TWENTY FIVE CENTS
$252.25
Ticket Void after 90 days ASSET# 23208

GoofyCoin

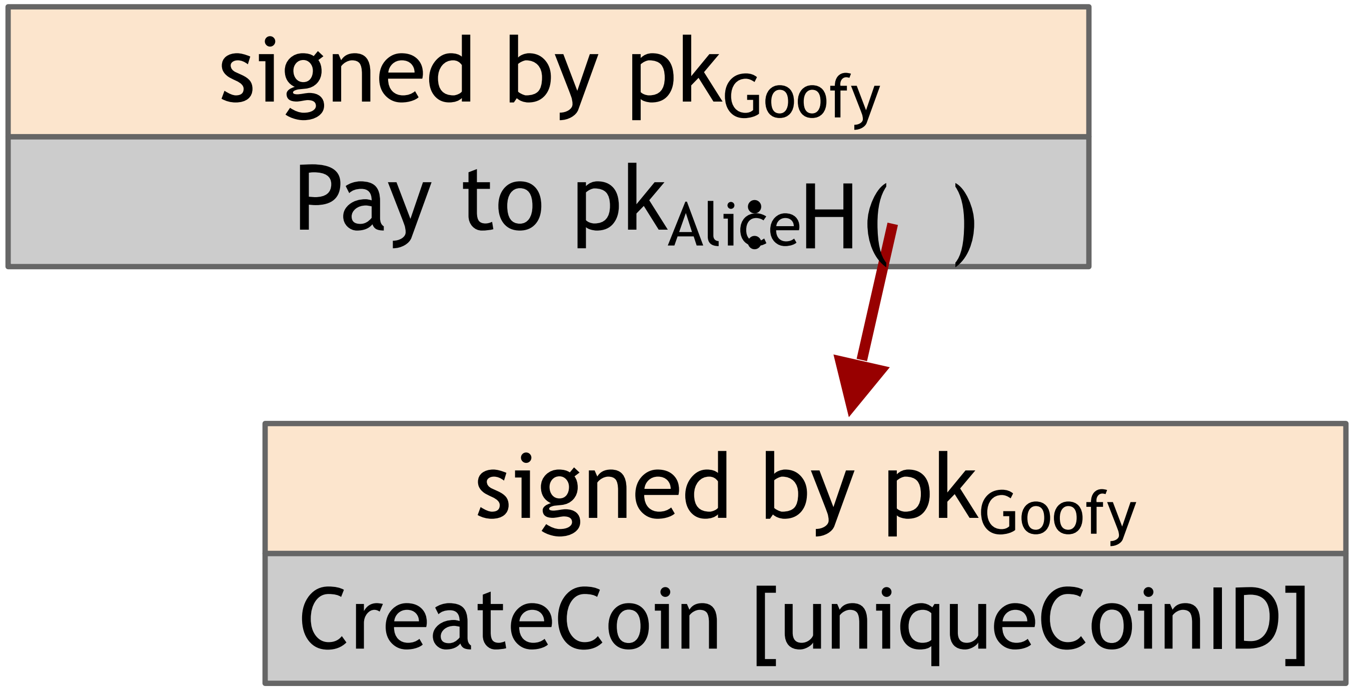
Goofy can create new coins

signed by pk_{Goofy}

CreateCoin [uniqueCoinID]

New coins belong to me.

A coin's owner can spend it.



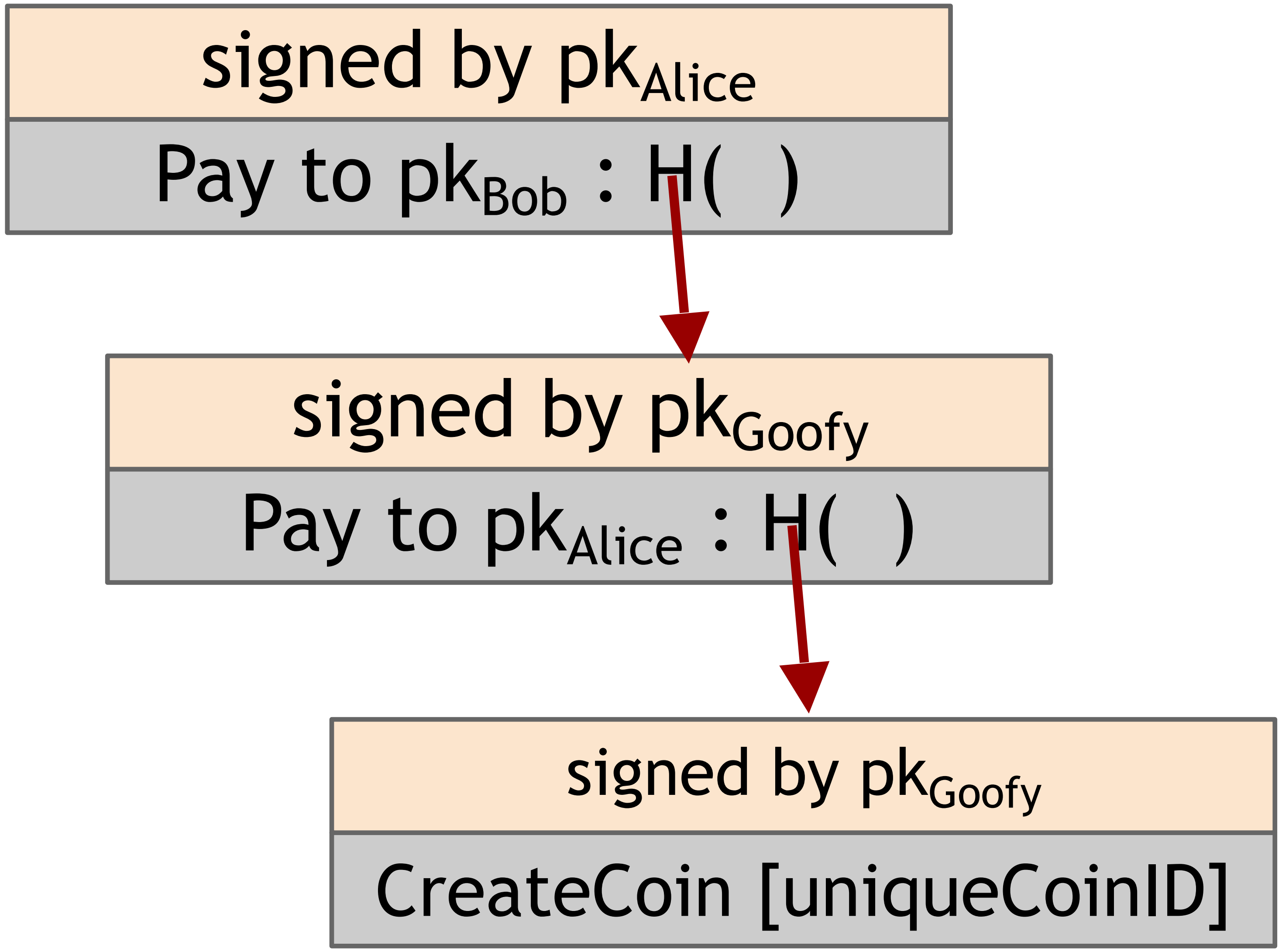
signed by pk_{Goofy}

Pay to pk_{Alice} : H( )

signed by pk_{Goofy}

CreateCoin [uniqueCoinID]

The recipient can pass on the coin again.



signed by pk_{Alice}

Pay to pk_{Bob} : H( )

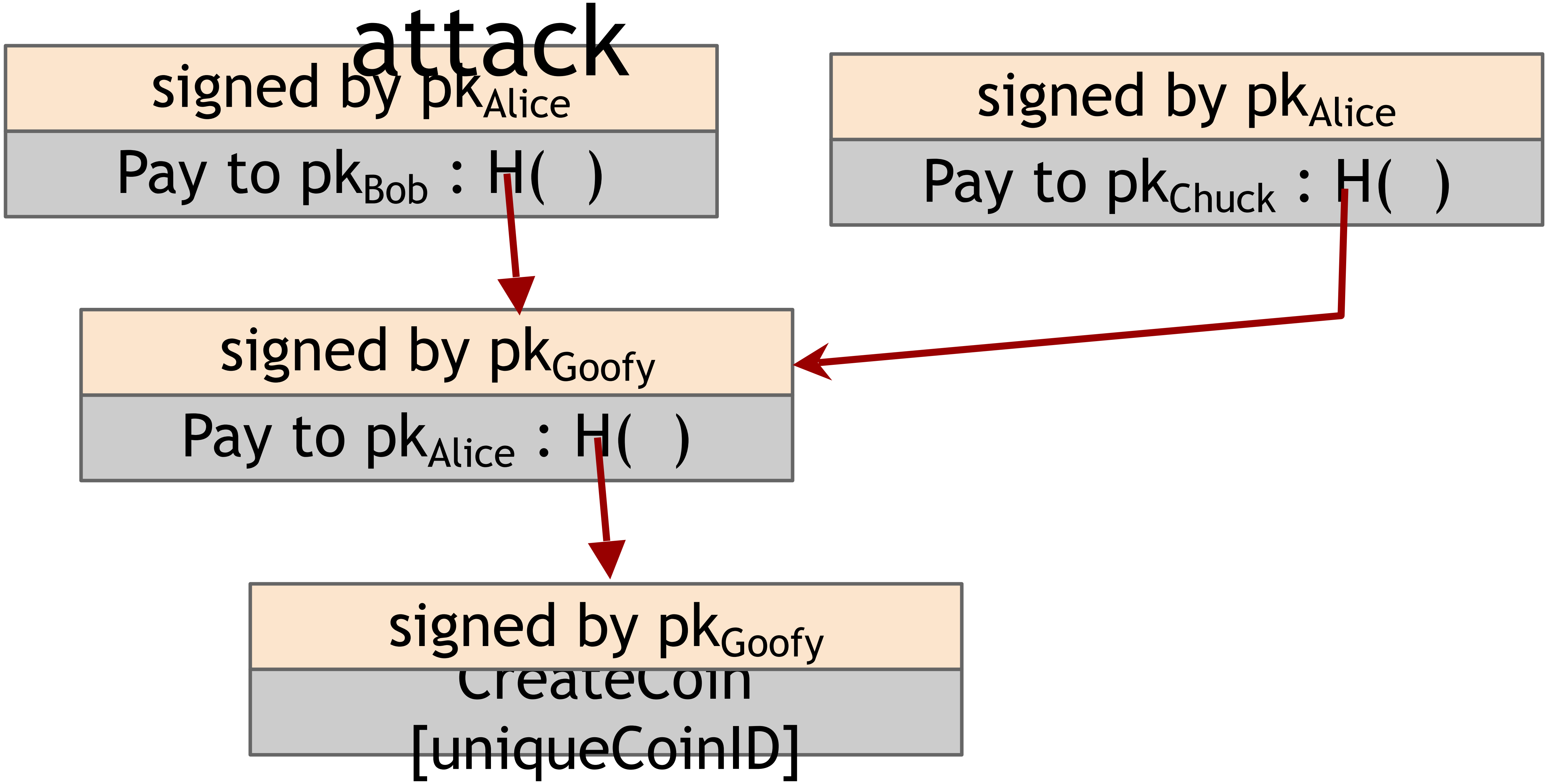
signed by pk_{Goofy}

Pay to pk_{Alice} : H( )

signed by pk_{Goofy}

CreateCoin [uniqueCoinID]

double-spending attack



signed by pk_{Alice}

Pay to pk_{Bob} : H( )

signed by pk_{Alice}

Pay to pk_{Chuck} : H( )

signed by pk_{Goofy}

Pay to pk_{Alice} : H( )

signed by pk_{Goofy}

CreateCoin [uniqueCoinID]

double-spending attack

the main design challenge in digital currency

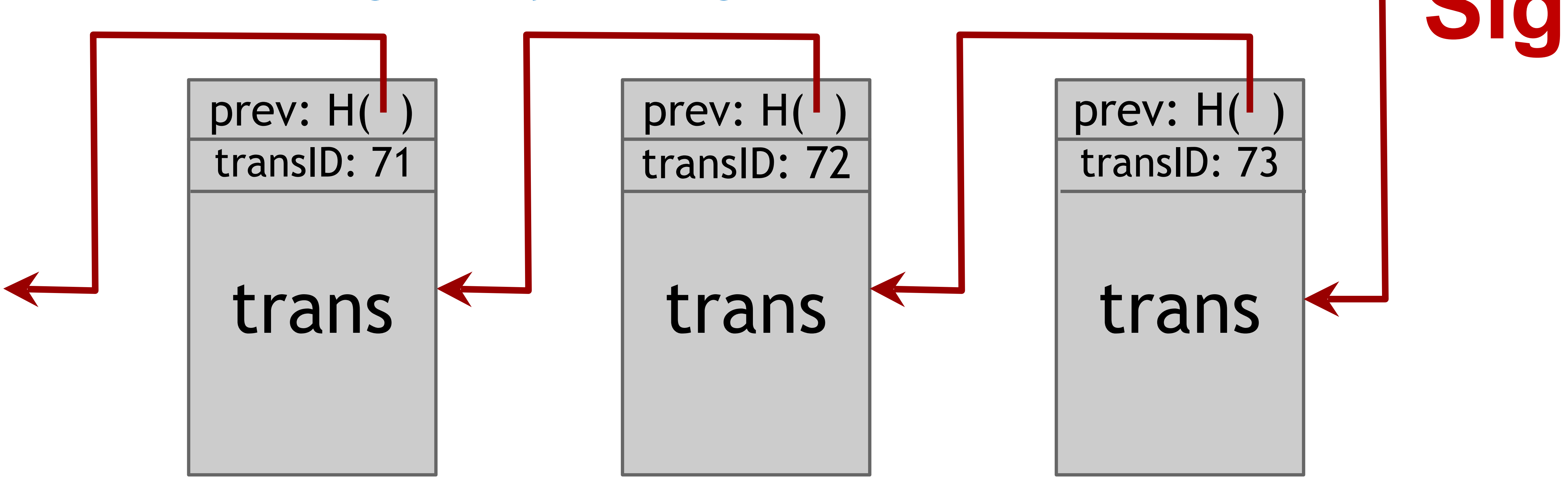
How do we solve this?

- Simplest answer: send all transactions to an atomic, append-only centralized ledger
- Have the ledger provide a definite ordering for transactions
  - If two transactions conflict, simply disallow the later one
- No TX is valid unless the ledger has "approved" and ordered it

ScroogeCoin

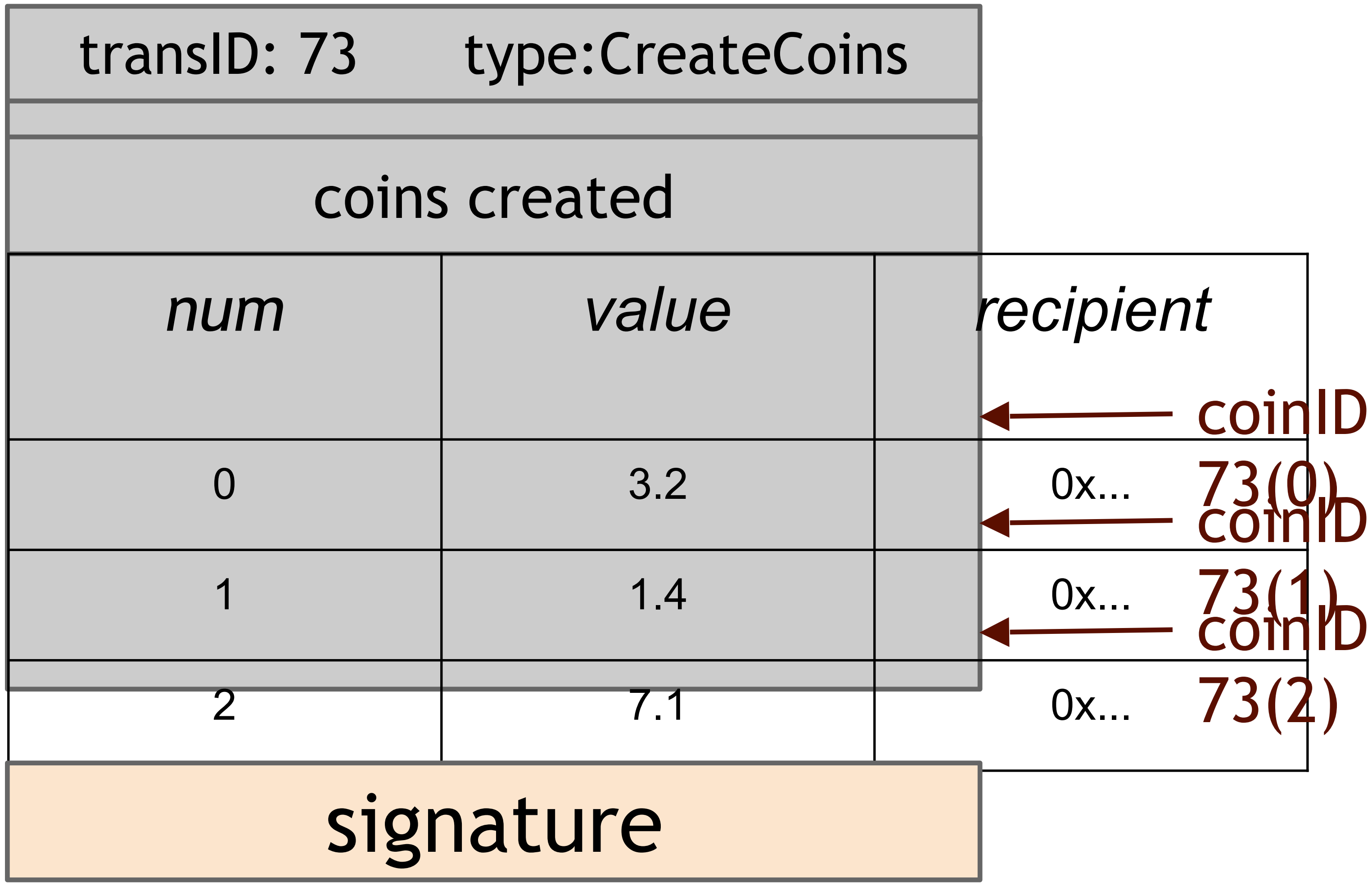
Scrooge publishes a history of all transactions in an “append-only” ledger

Implement the ledger using a block chain, signed by Scrooge



optimization: put multiple transactions in the same block

CreateCoins transaction creates new coins



transID: 73 type:CreateCoins

coins created

num	*value*	*recipient*
0	3.2	0x...
1	1.4	0x...
2	7.1	0x...

signature

coinID 73(0)

coinID 73(1)

coinID 73(2)

Valid, because I said so.

PayCoins transaction consumes (and destroys) some coins,
and creates new coins of the same total value

transID: 73 type:PayCoins

consumed coinIDs:
68(1), 42(0), 72(3)

coins created

num	*value*	*recipient*
0	3.2	0x...
1	1.4	0x...
2	7.1	0x...

signatures

Valid if:

- -- consumed coins valid,
- -- not already consumed,
- -- total value out = total value in, and
- -- signed by owners of all consumed coins

Immutable coins

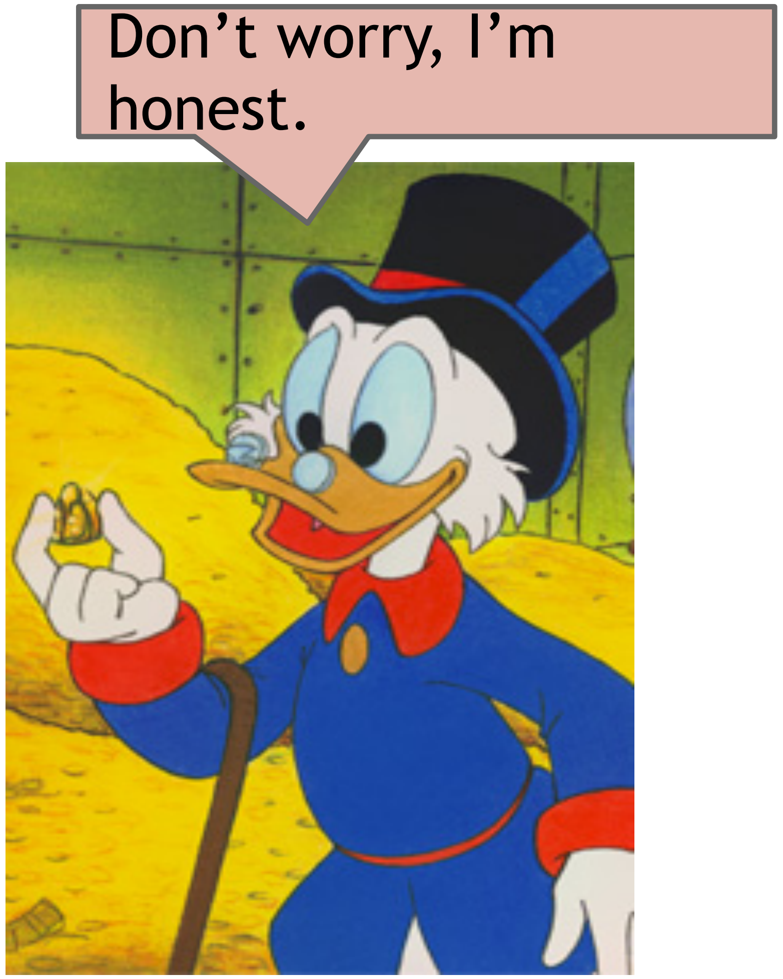
Coins can’t be transferred, subdivided, or combined.

But: you can get the same effect by using transactions

to subdivide: create new trans

consume your coin

pay out two new coins to yourself



Crucial question:

Can we descroogify the currency, and operate without any central, trusted party?

Centralization vs. Decentralization

- **Competing paradigms that underlie many technologies**
- Decentralized != Distributed
  (as in distributed system) but we'll often use them as synonyms

Centralization vs. Decentralization

- Examples:
  - email?
  - WWW?
  - DNS?
- What about software development?

Aspects of decentralization in Bitcoin

1. Who maintains the ledger?
2. Who has authority over which transactions are valid?
3. Who creates (and obtains) new bitcoins?
4. Who determines how the rules change?
5. How do these coins acquire monetary value?

Aspects of decentralization in Bitcoin

Peer-to-peer network:

open to anyone, low barrier to entry

Mining:

open to anyone, but inevitable concentration of power
often seen as undesirable

Updates to software:

core developers trusted by community, have great power

Distributed consensus

Bitcoin’s key challenge

Key technical challenge of decentralized
e-cash: distributed consensus

or: how to decentralize ScroogeCoin

Why consensus protocols?

Traditional motivation: reliability in distributed systems

Distributed key-value store enables various applications:
DNS, public key directory, stock trades …

Good targets for Altcoins!

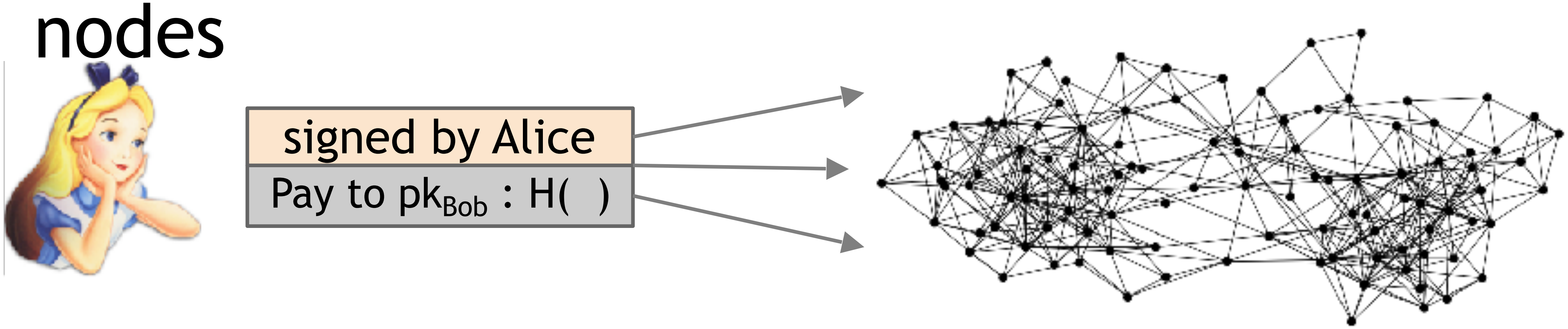
Defining distributed consensus

The protocol terminates and all honest nodes decide on the same value

This value must have been proposed by some honest node

Bitcoin is a peer-to-peer system

When Alice wants to pay Bob:
she broadcasts the transaction to all Bitcoin nodes



signed by Alice

Pay to pk_{Bob} : H( )

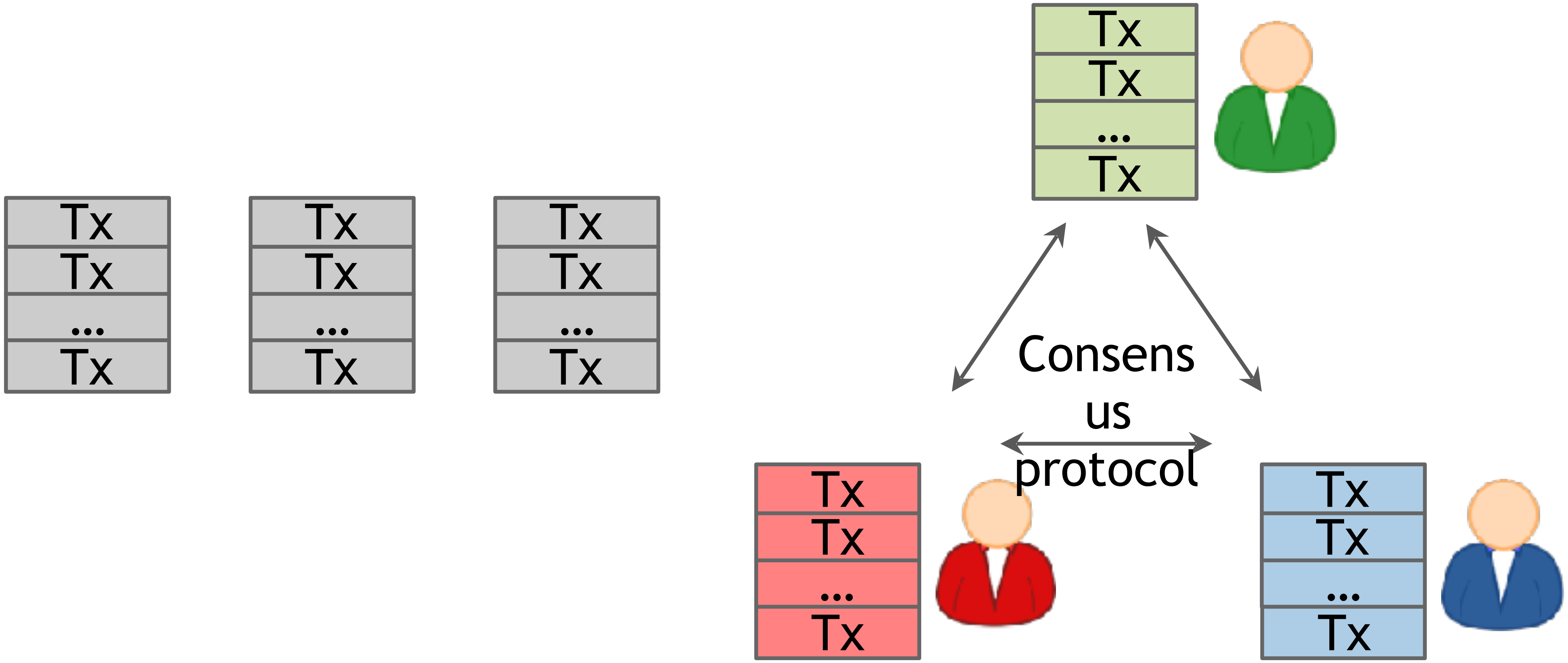
Note: Bob's computer is not in the picture

How consensus could work in Bitcoin

At any given time:

- All nodes have a sequence of blocks of transactions they've reached consensus on
- Each node has a set of outstanding transactions it's heard about

How consensus could work in Bitcoin



OK to select any valid block, even if proposed by only one node

Why consensus is hard

Nodes may crash
Nodes may be malicious

Network is imperfect

- Not all pairs of nodes connected
- Faults in network
- Latency

No notion of global time