

# Lightning Network

Joseph Poon     joseph@lightning.network  
Thaddeus Dryja   rx@awsomnet.org

# Problems

- Transactions aren't instant
- Micropayments don't actually work
  - High transaction fees
- “Bitcoin Doesn't Scale”

# “Bitcoin Doesn’t Scale”

- 1 MB blocks:
  - 7 transactions per second @ 250 bytes/tx
  - ~220 million transactions per **year**
  - Not enough for a city, let alone the world
- 1 Billion transactions per day:
  - 1.6 GB blocks (1655 MB)
  - 87 Terabytes/year (87029089 MB)
  - Maybe enough for one large metro area?
  - Centralization (mining!)

# “Bitcoin Doesn’t Scale”

- 7 billion people doing 2 blockchain transactions per day
  - 24 GB blocks
  - 3.5 TB/day
  - 1.27 PB/year
- Bigger blocks = Centralization
  - Very few full nodes
  - Very few miners
  - De facto inability to validate blockchain

# Scalability Solutions

- The SQL Database Model
  - Very scalable, very fast
  - Off chain transactions implemented today with ChangeTip, Coinbase, others
- Sidechains
  - Many blockchains with inter-chain transfers
- Payment Channels
  - Many payments between two pre-determined parties

# Scalability Solutions

- The SQL Database Model
  - Also implemented by MTGox Redeemable Codes
- Sidechains
  - Not primarily a scalability solution
  - Sending funds between chains is two transactions
- Payment Channels
  - Only helps when people pay each other many times (recurring billing, time-based microtransactions)

# Anyone to Anyone Payments

- In bitcoin, any output can pay to any other
- In the SQL database model:
  - I need to have an entry in your SQL db
- On Sidechains:
  - I need to be on your sidechain
- In a Payment Channel:
  - I need to have a channel open with you

# Bitcoin Lightning Network

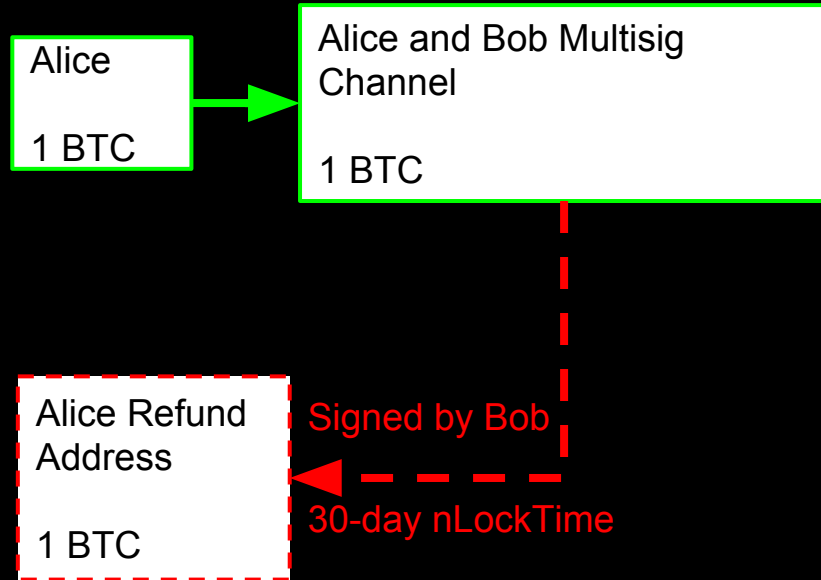
- Payment channels between many parties in a multi-hop hub and spoke model (similar to internet routing)
- Minimally trusted intermediaries (they can't take your coins)
- With a malleability fix via a soft-fork, Bitcoin can scale to billions of transactions per day



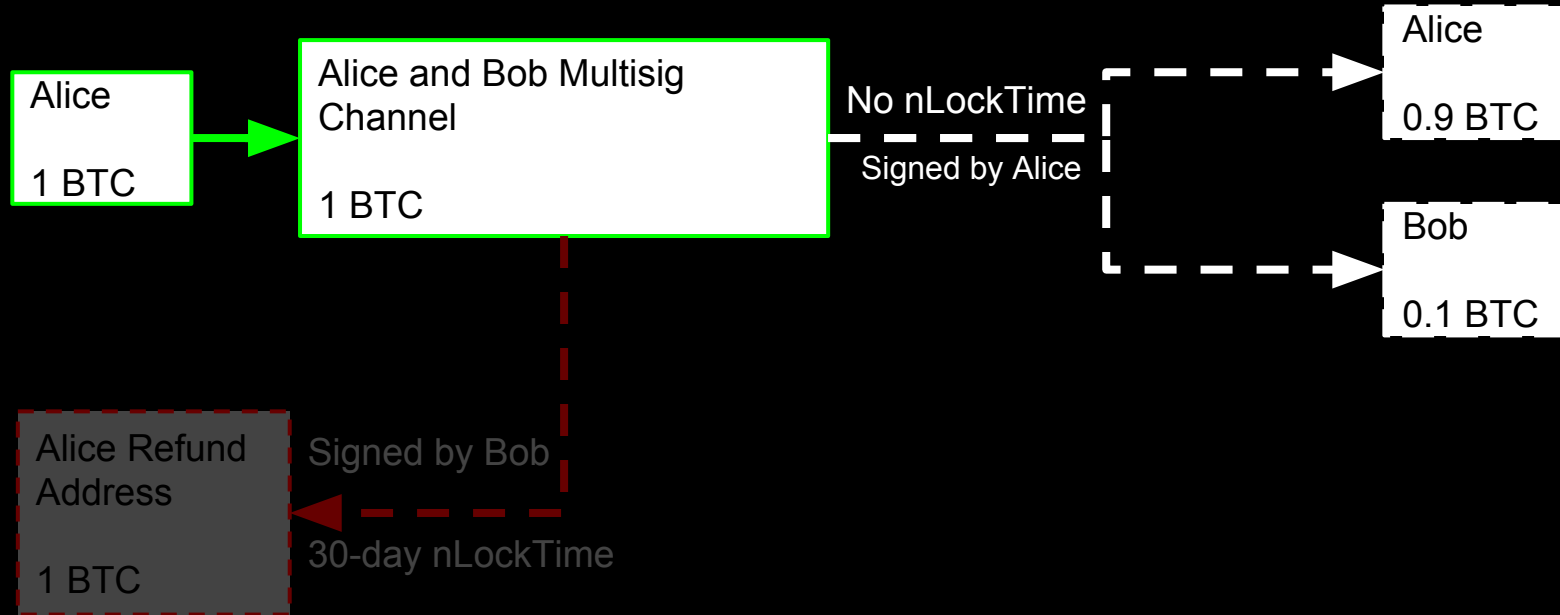
# What are Payment Channels

- Introduced a while ago (not a new idea)
- Uses multi-sig
- Allows two people to send transactions to each other without hitting the Bitcoin blockchain

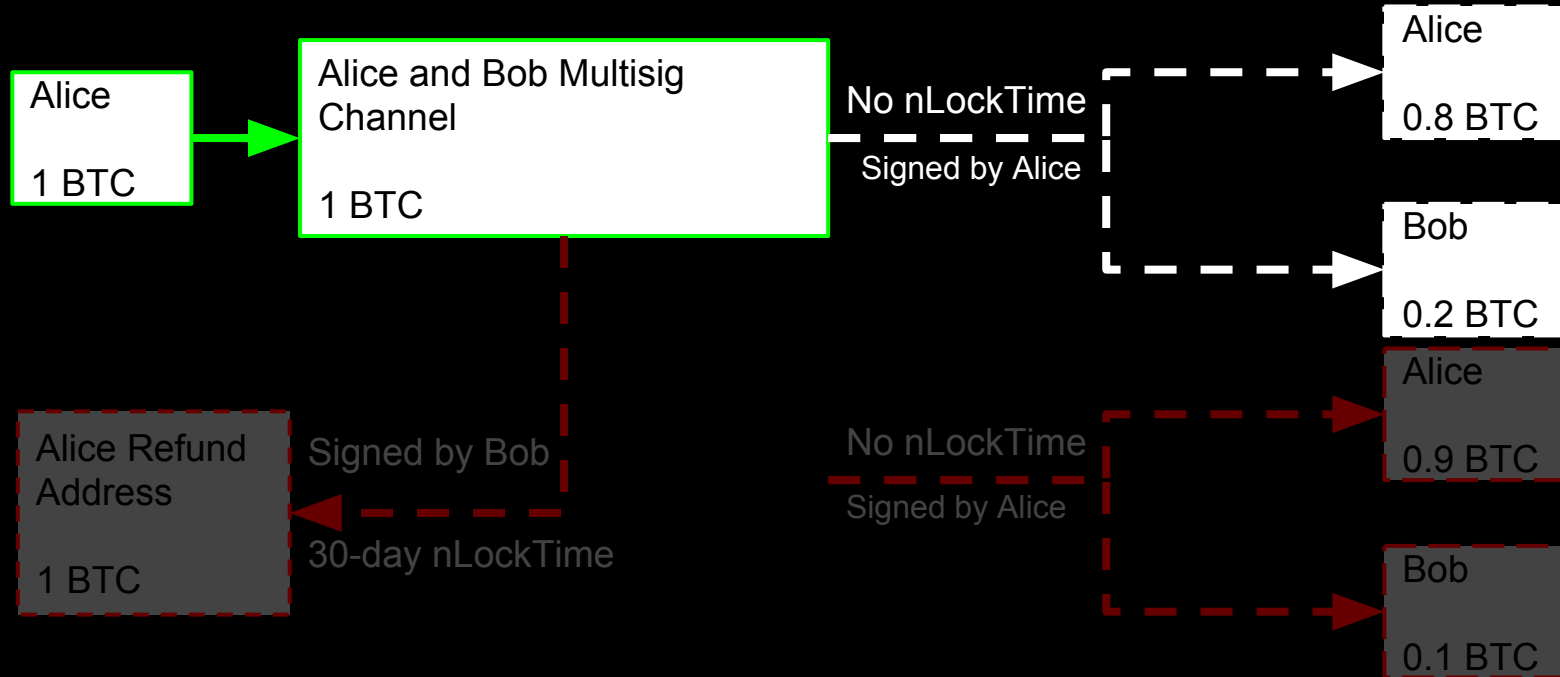
# Unidirectional Channel



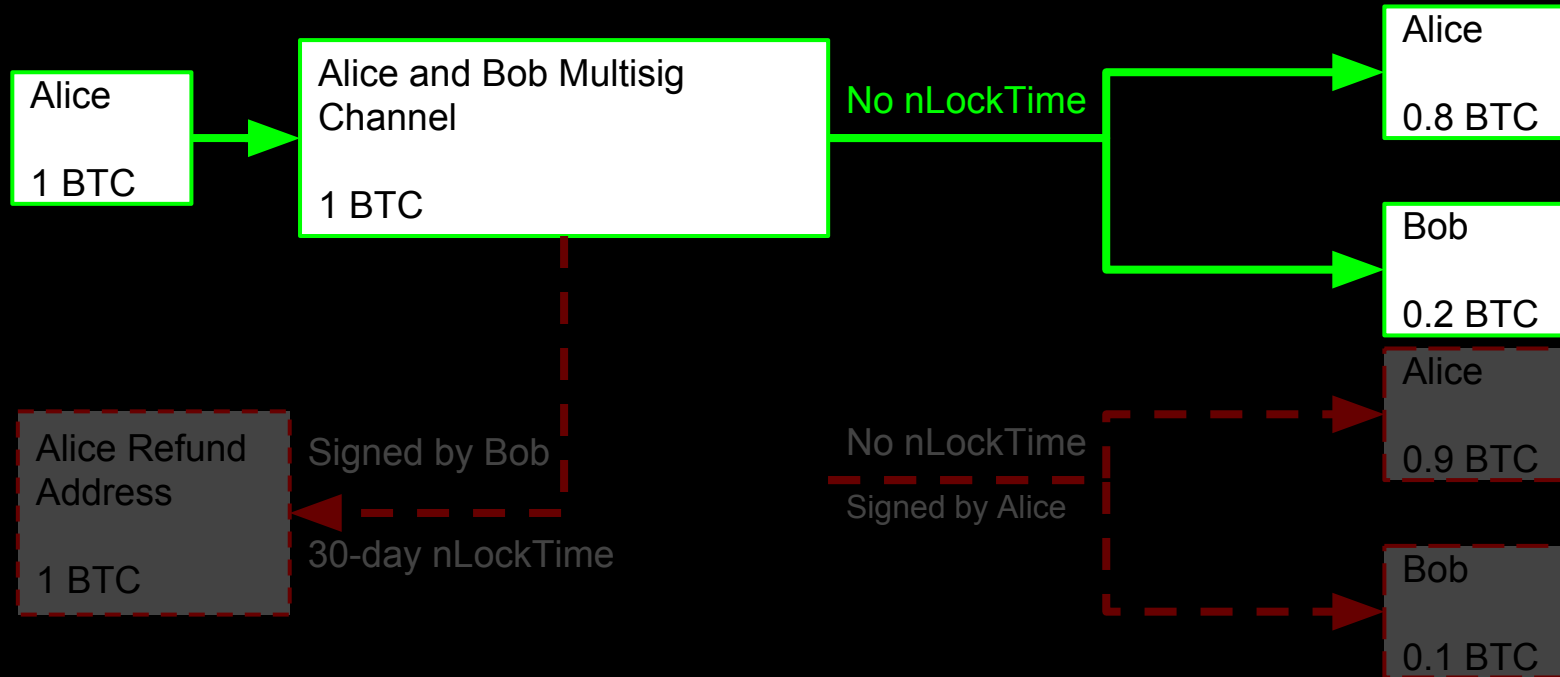
# Unidirectional Channel



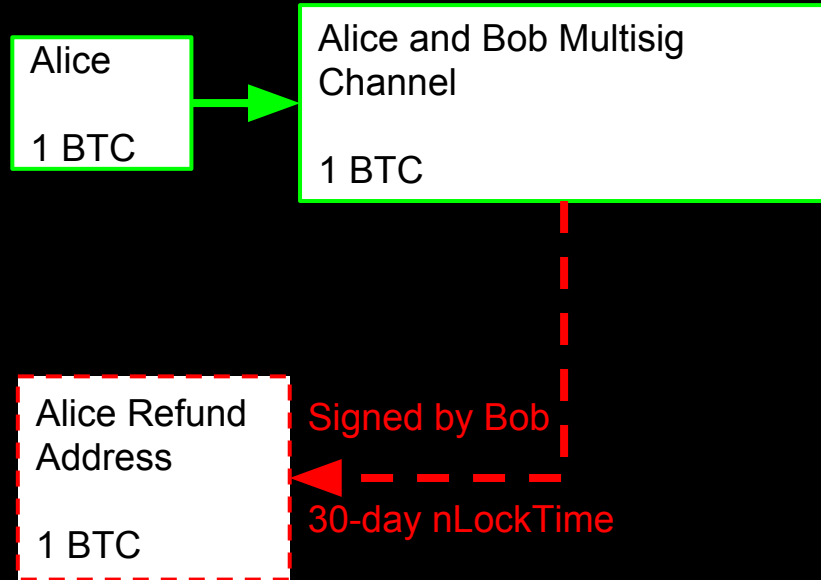
# Unidirectional Channel



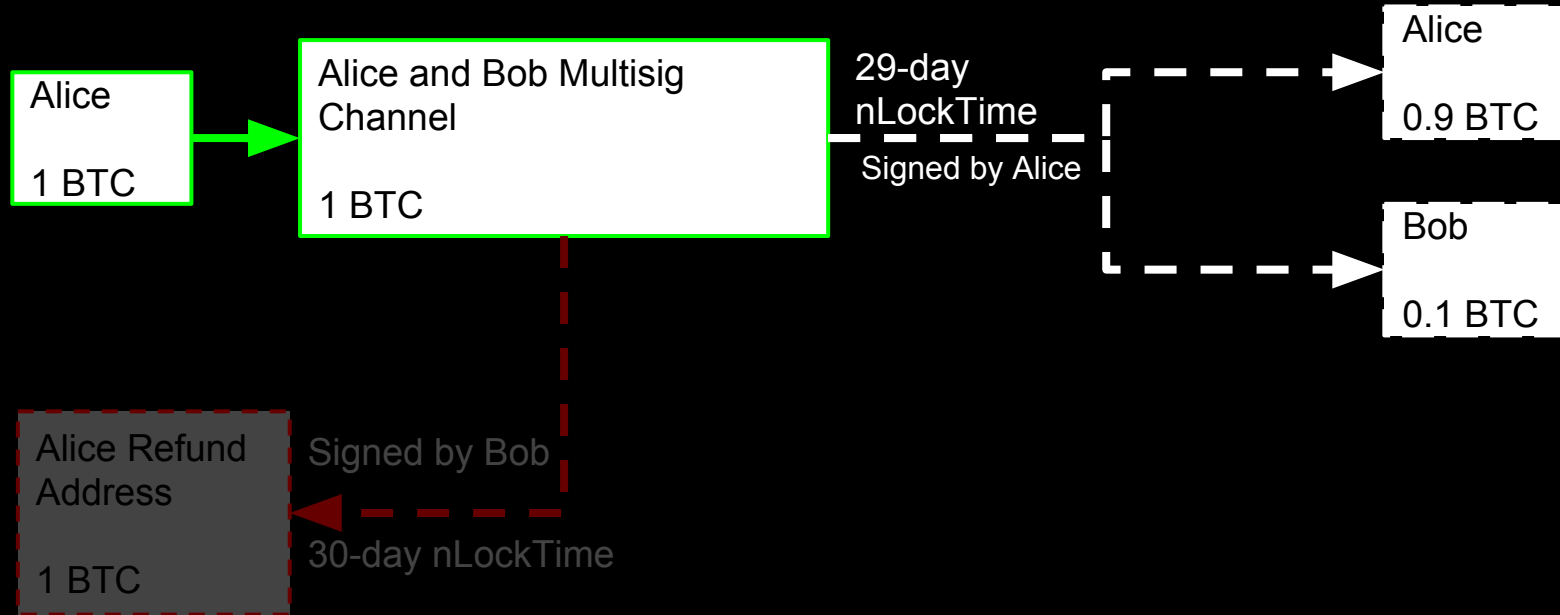
# Unidirectional Channel



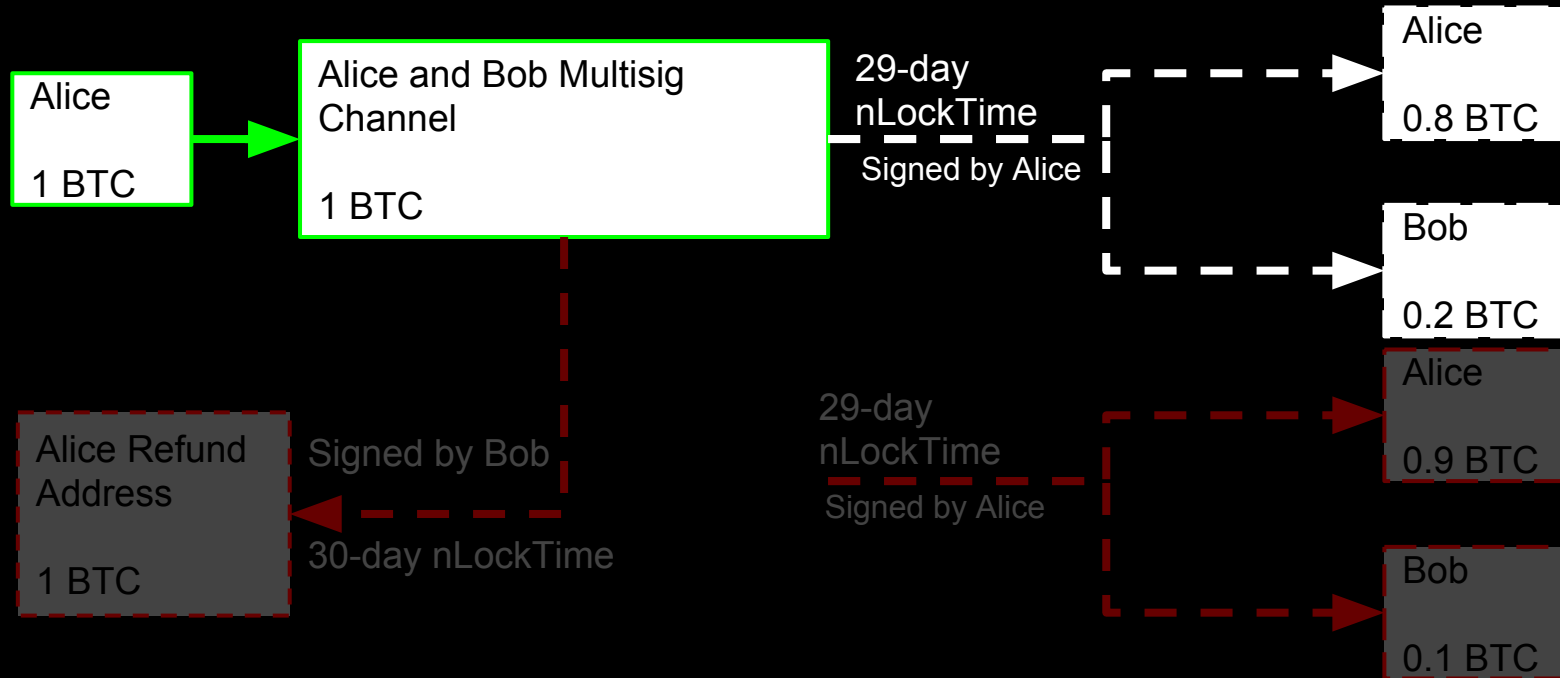
# Bidirectional Channel



# Bidirectional Channel - Payment

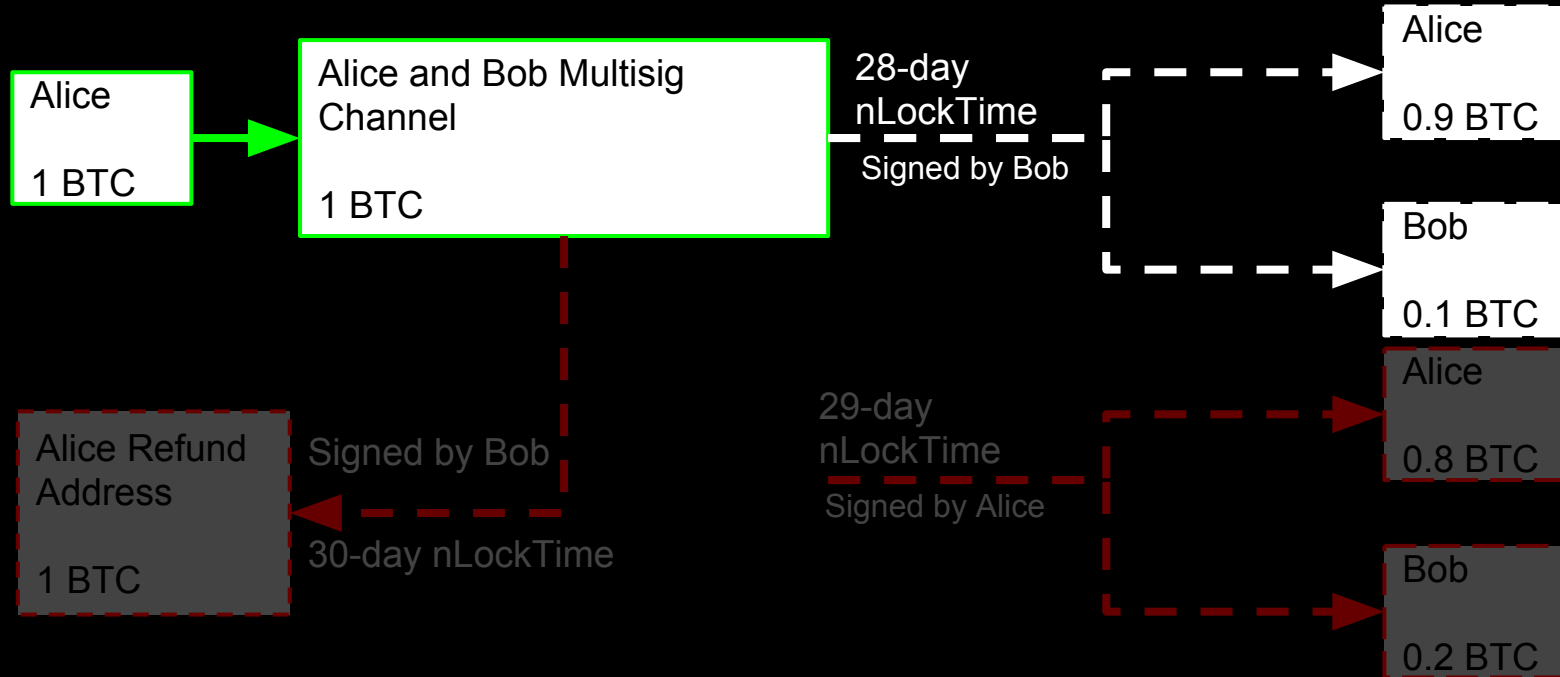


# Bidirectional Channel - Payment

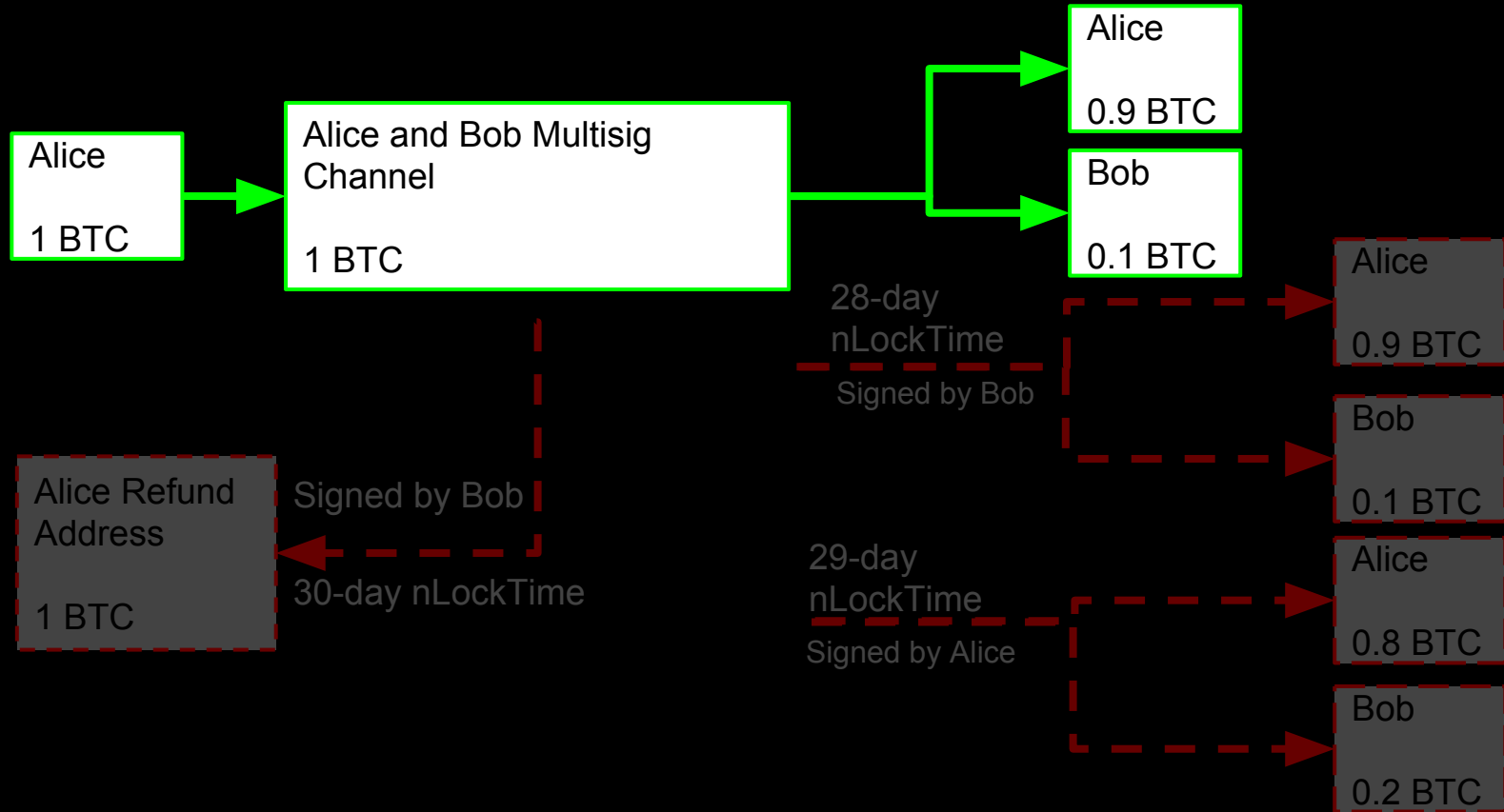




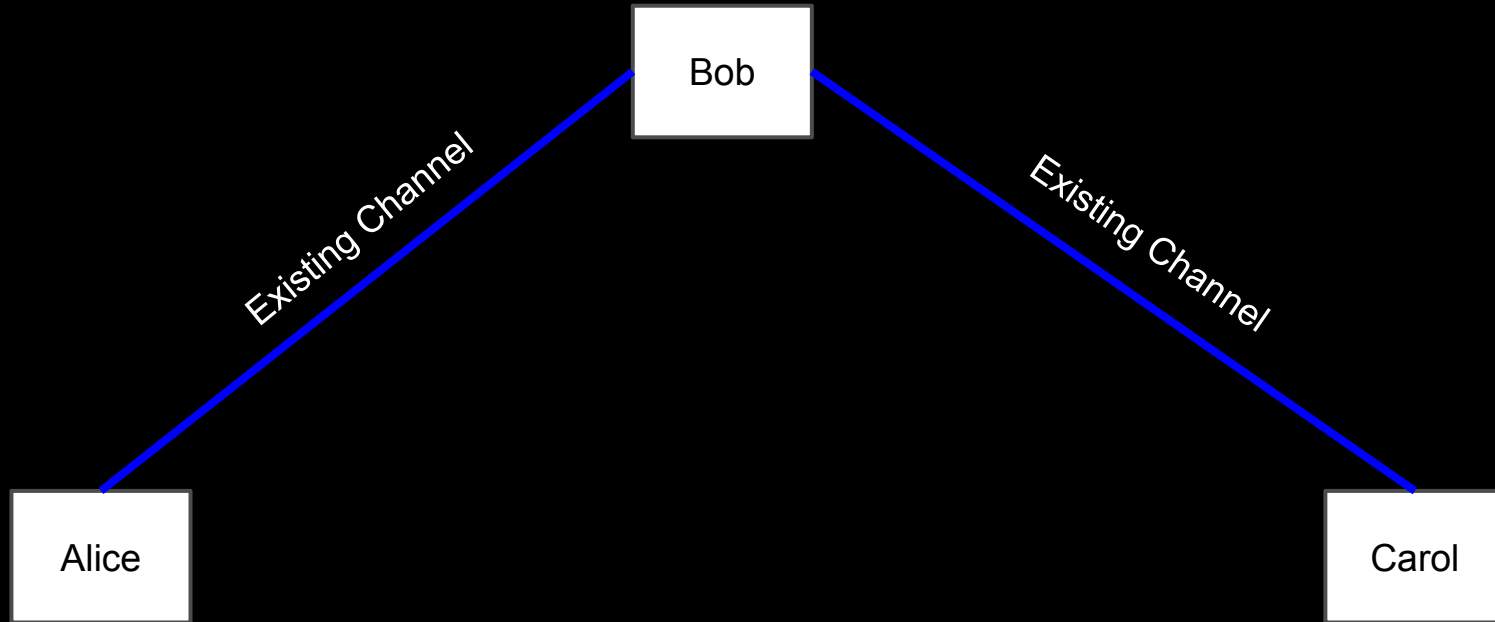
# Reversing Direction



# Closing Bidirectional Channel

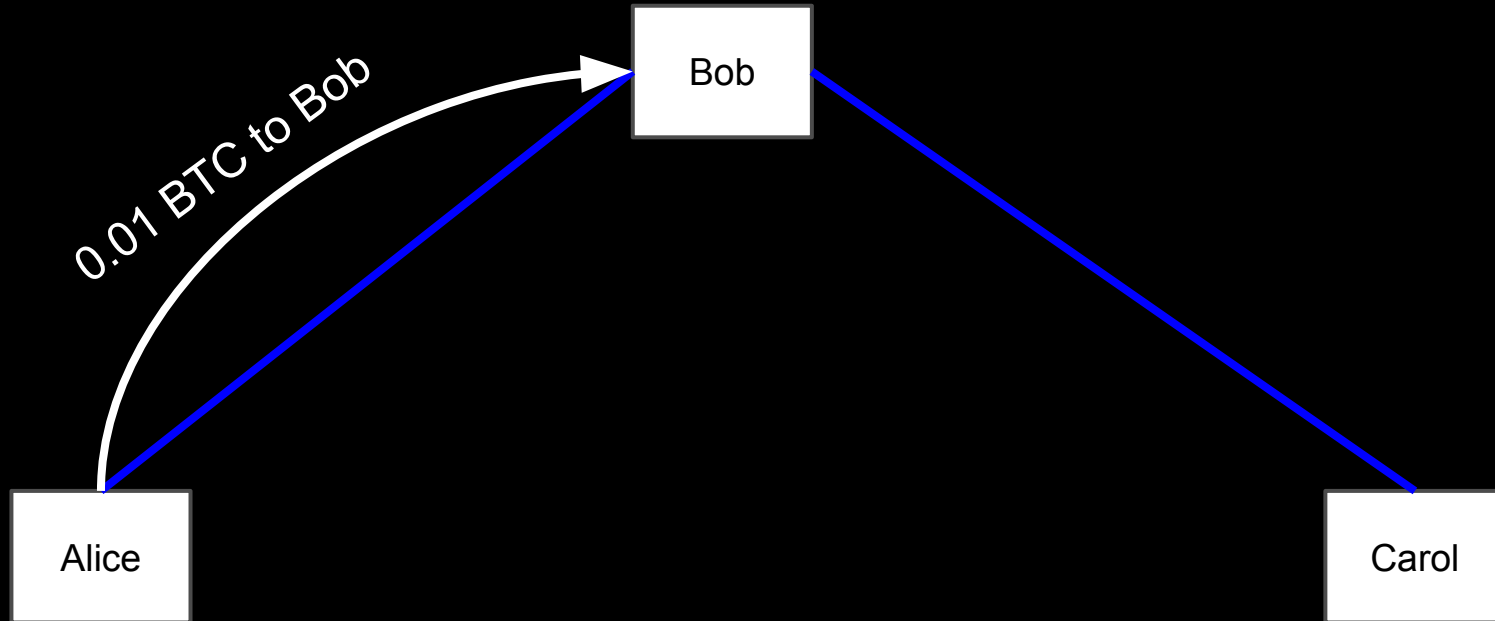


# 3 Party Payments

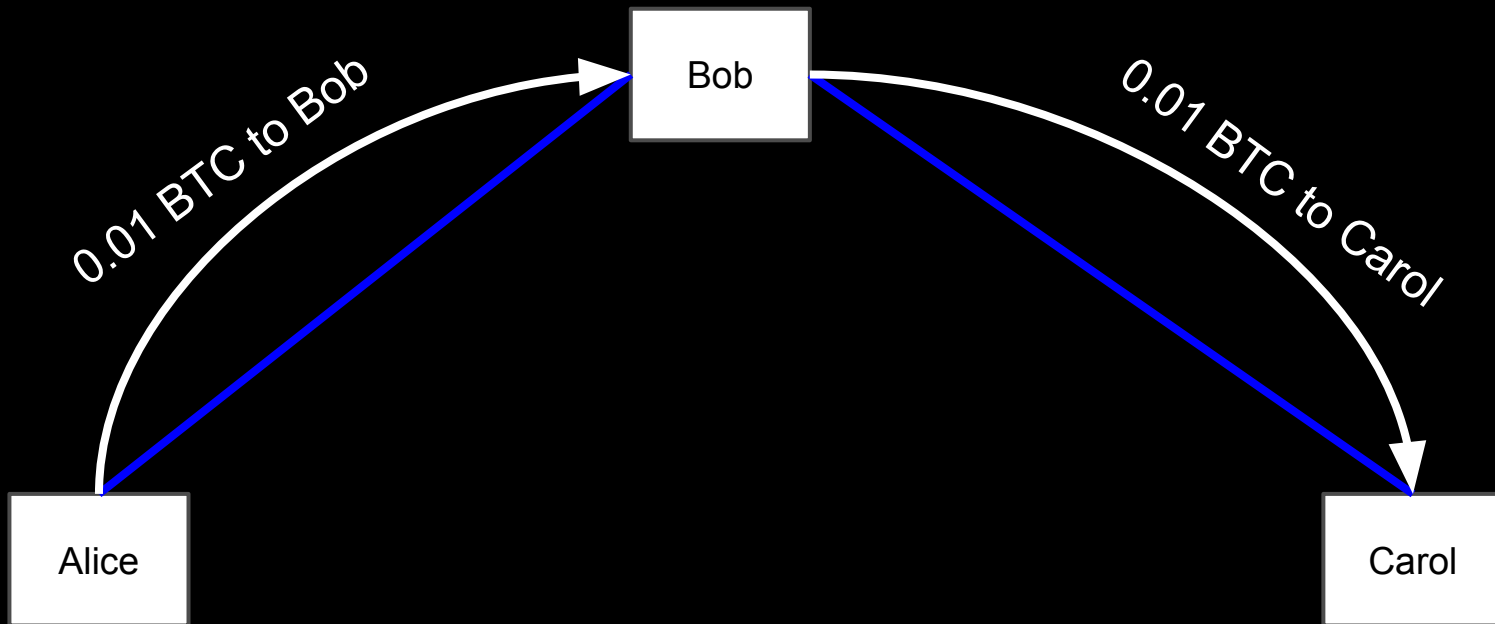


Alice wants to pay Carol, they both have a channel open with Bob

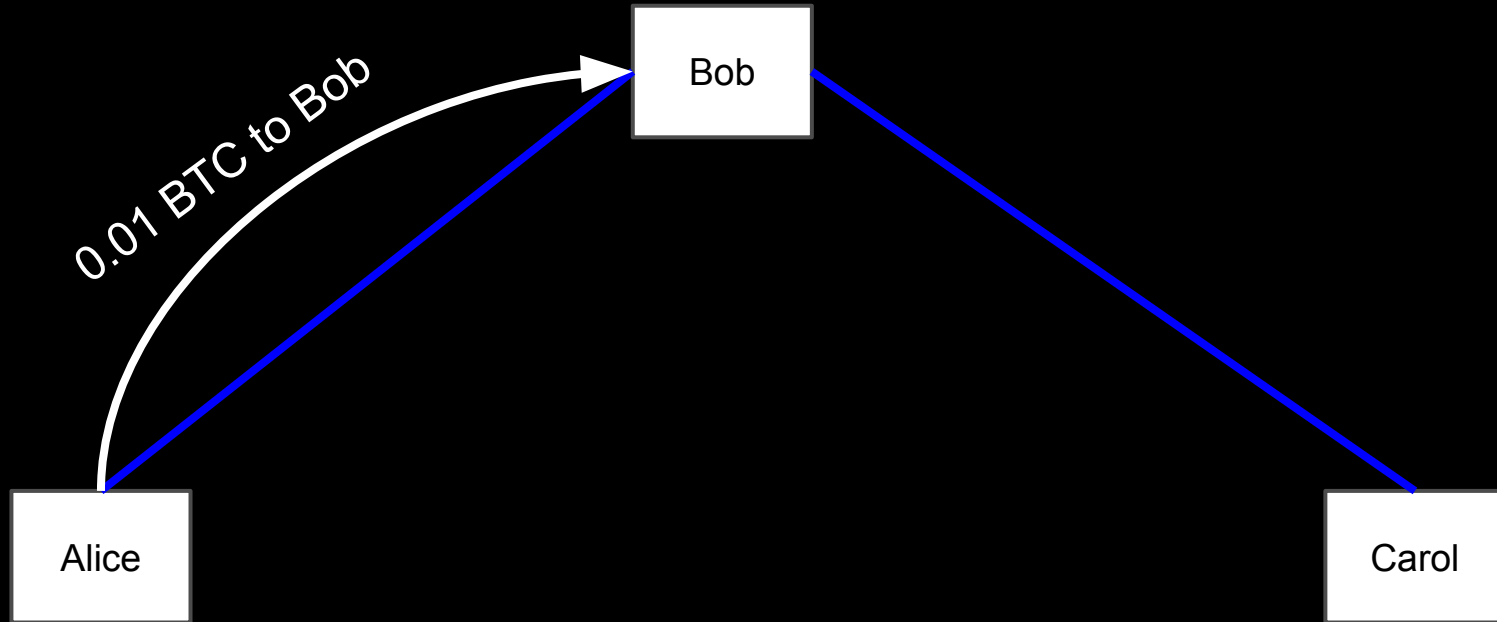
# 3 Party Payments



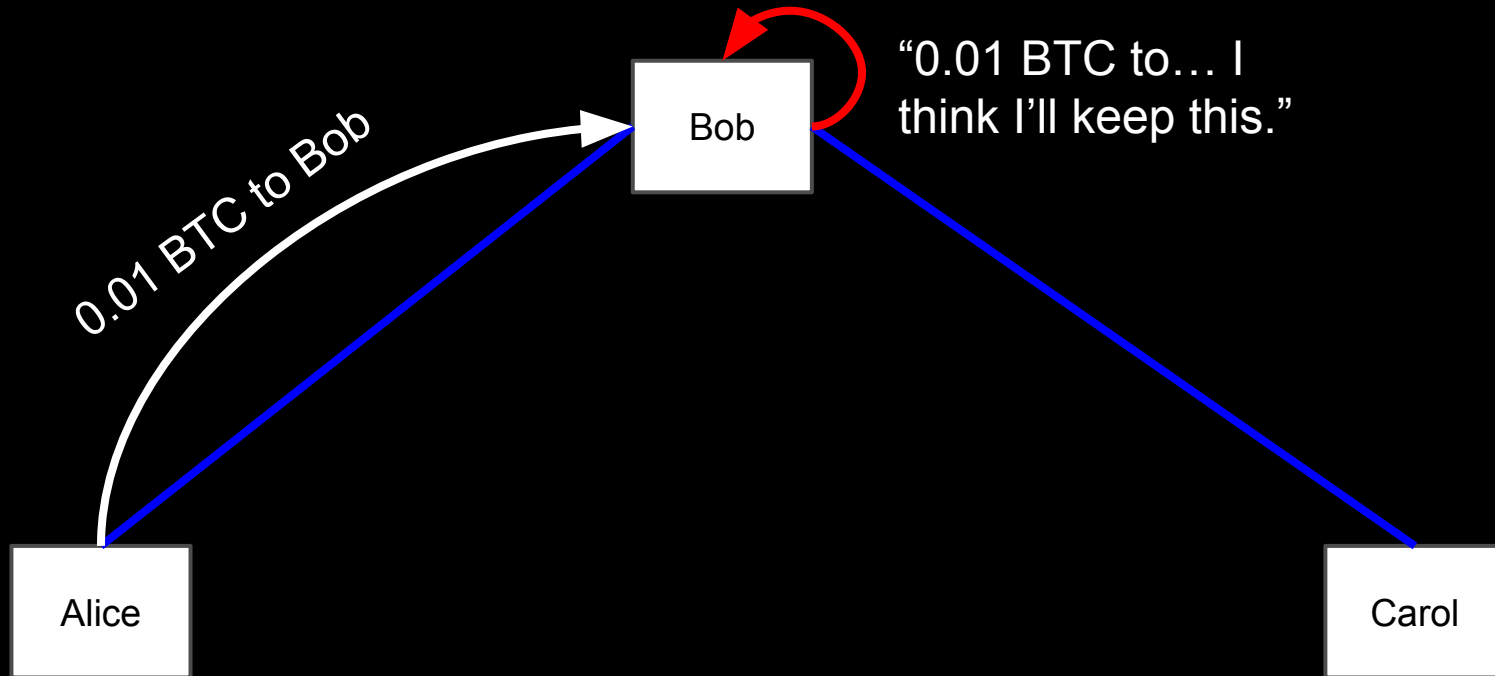
# 3 Party Payments



# 3 Party Payments - Trust Issues



# 3 Party Payments - Trust Issues



Problem: Bob can simply keep the 0.01 BTC  
Problem: Carol can claim she never got the coins!

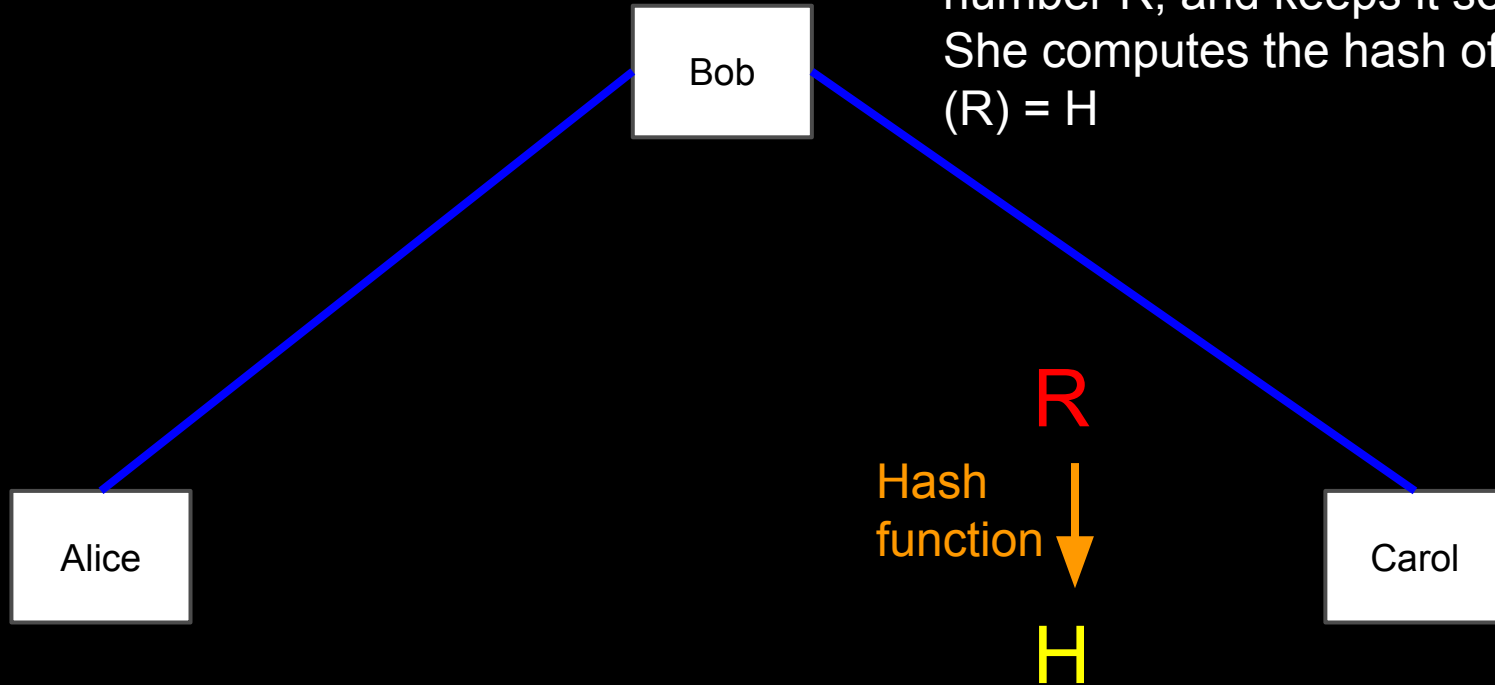
# Hash-Locked Contracts

- Using one-way hash functions, Alice can prove she sent funds to Carol off-chain
- Pay to Contract
  - Knowledge of secret  $R$  hashed into hash  $H$  proves receipt
  - Receiver signs a contract stating if  $R$  is disclosed funds have been received

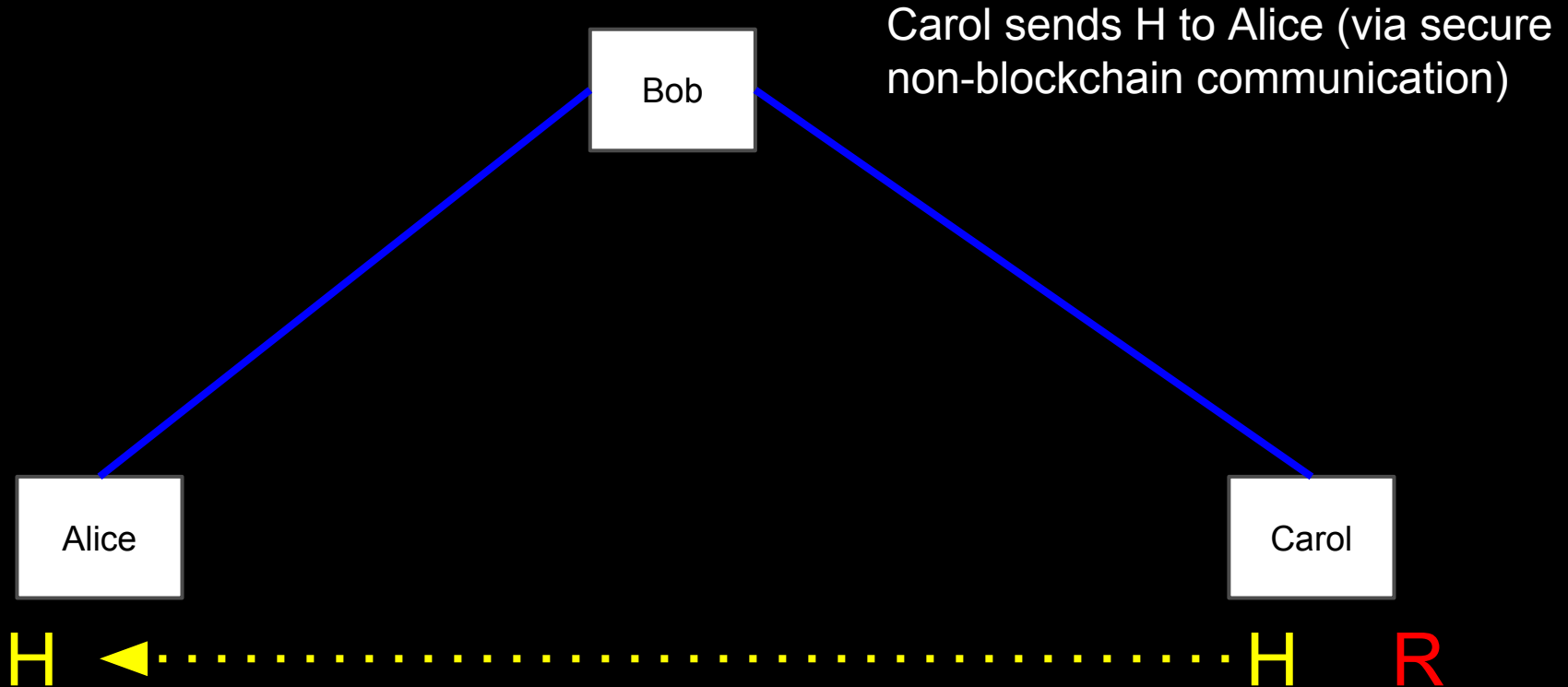


# Hash-Locked Contracts

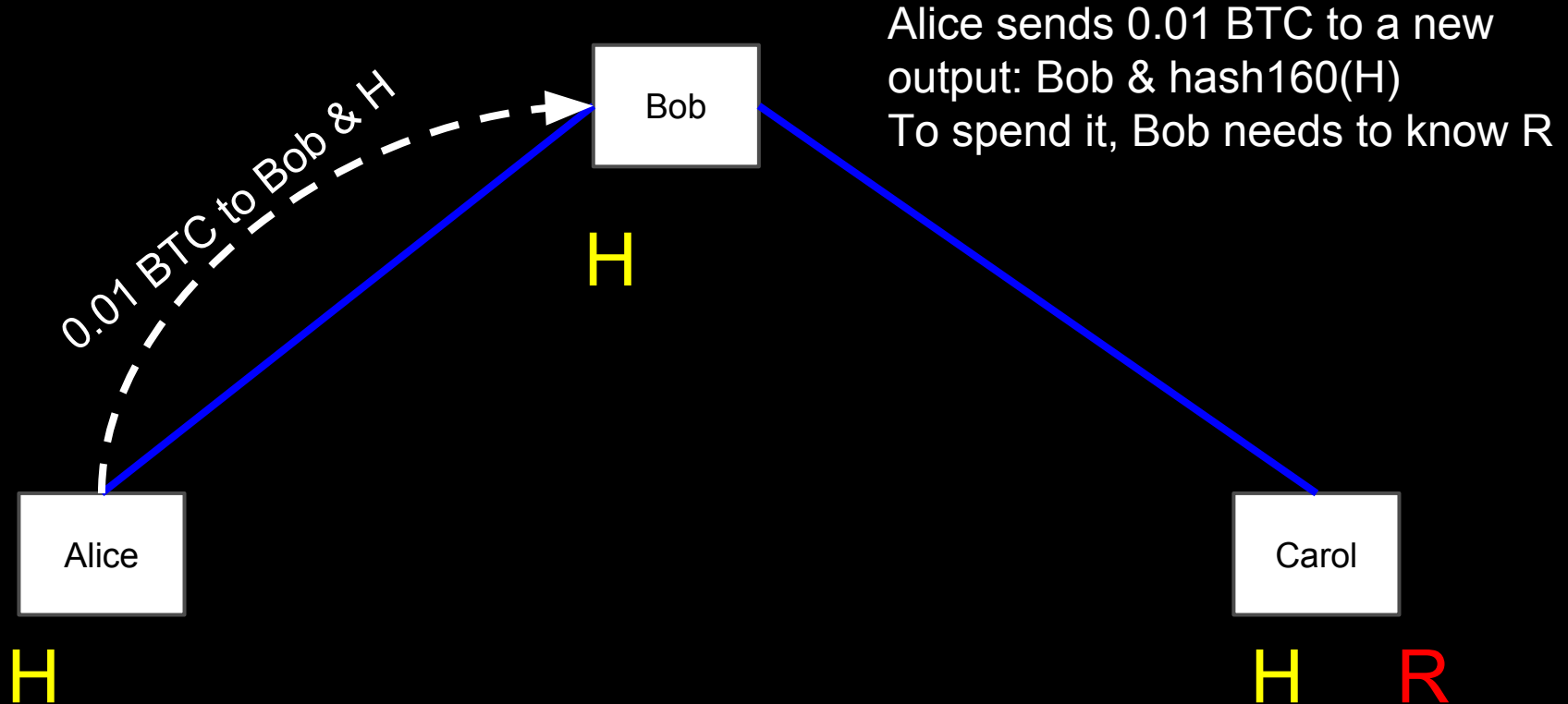
Carol makes a random number  $R$ , and keeps it secret. She computes the hash of  $R$ ,  $\text{hash}(R) = H$



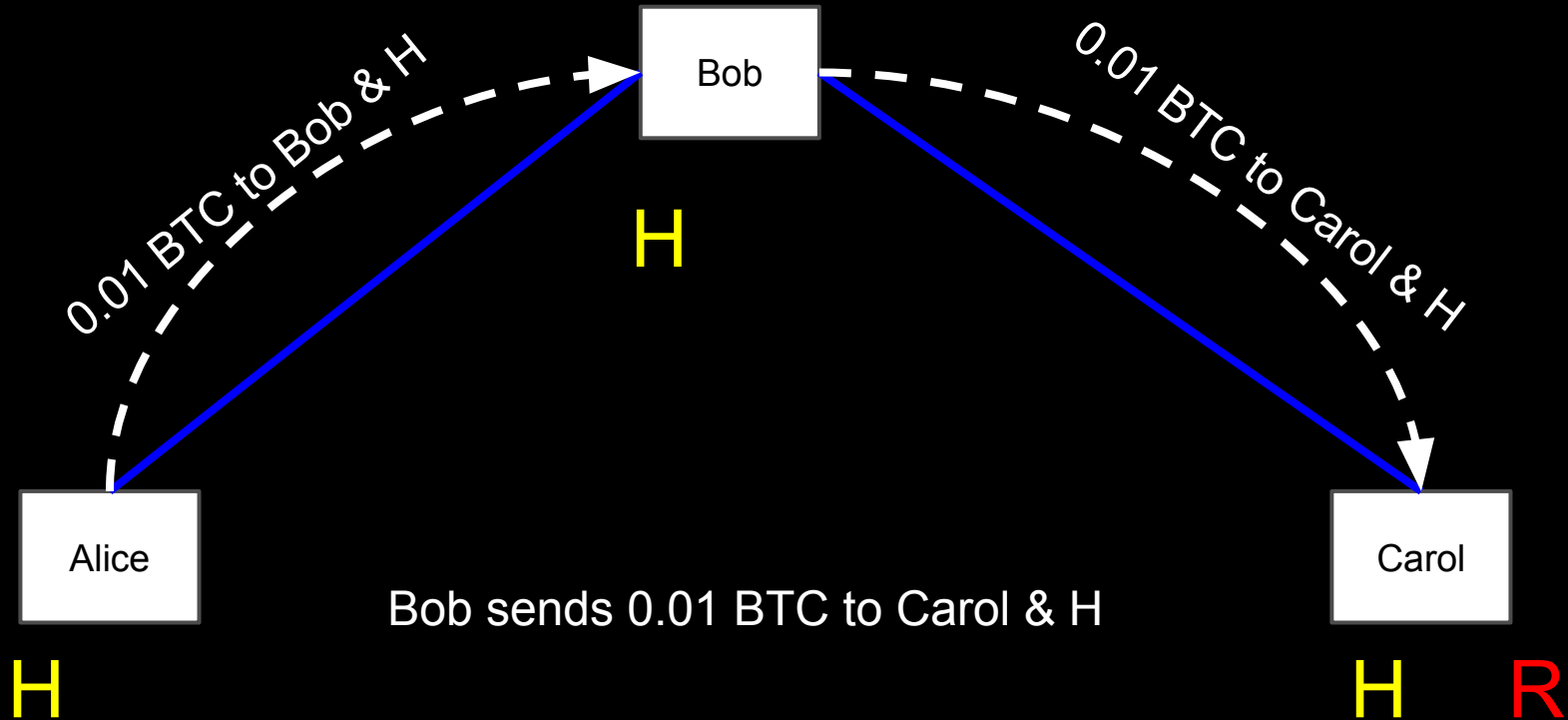
# Hash-Locked Contracts



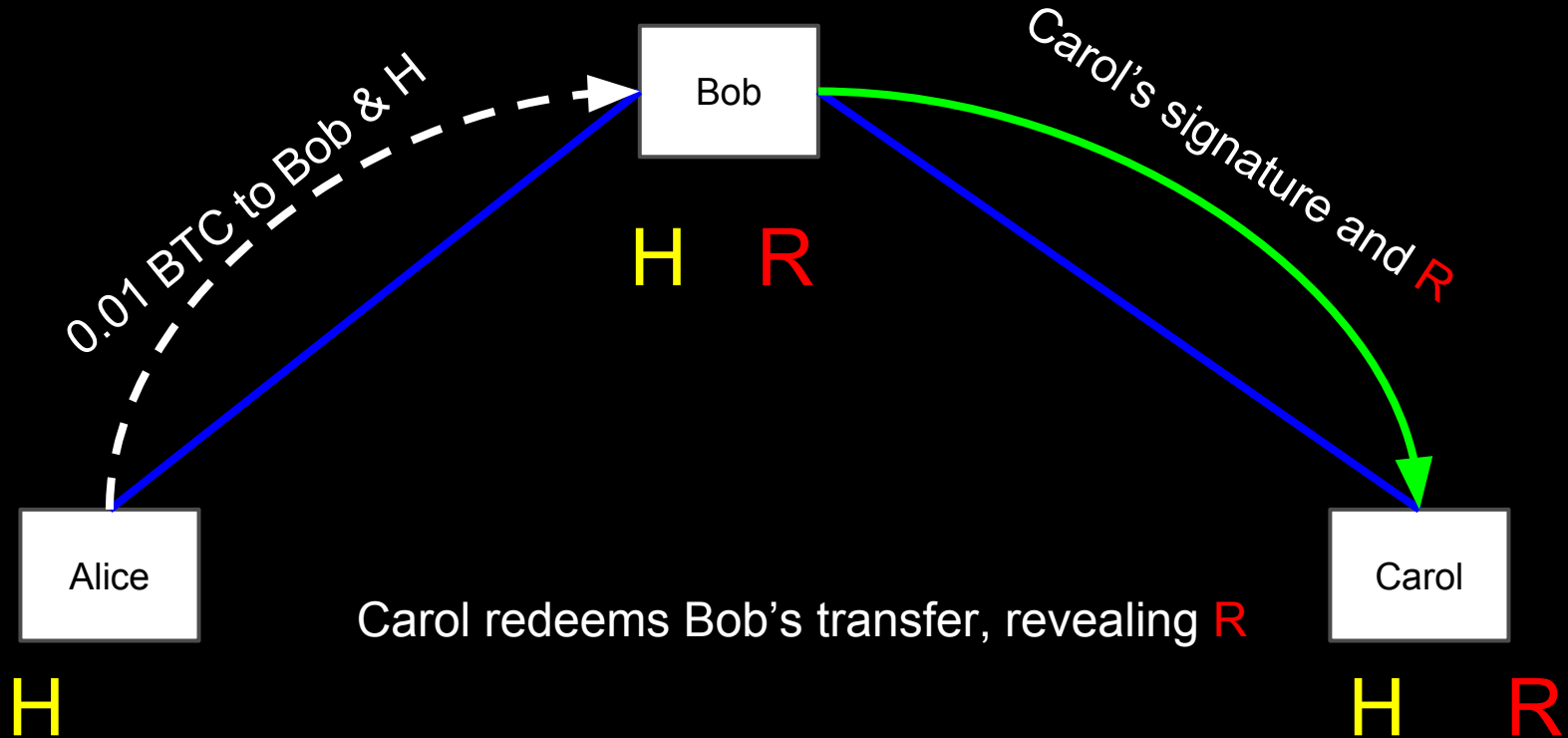
# Hash-Locked Contracts



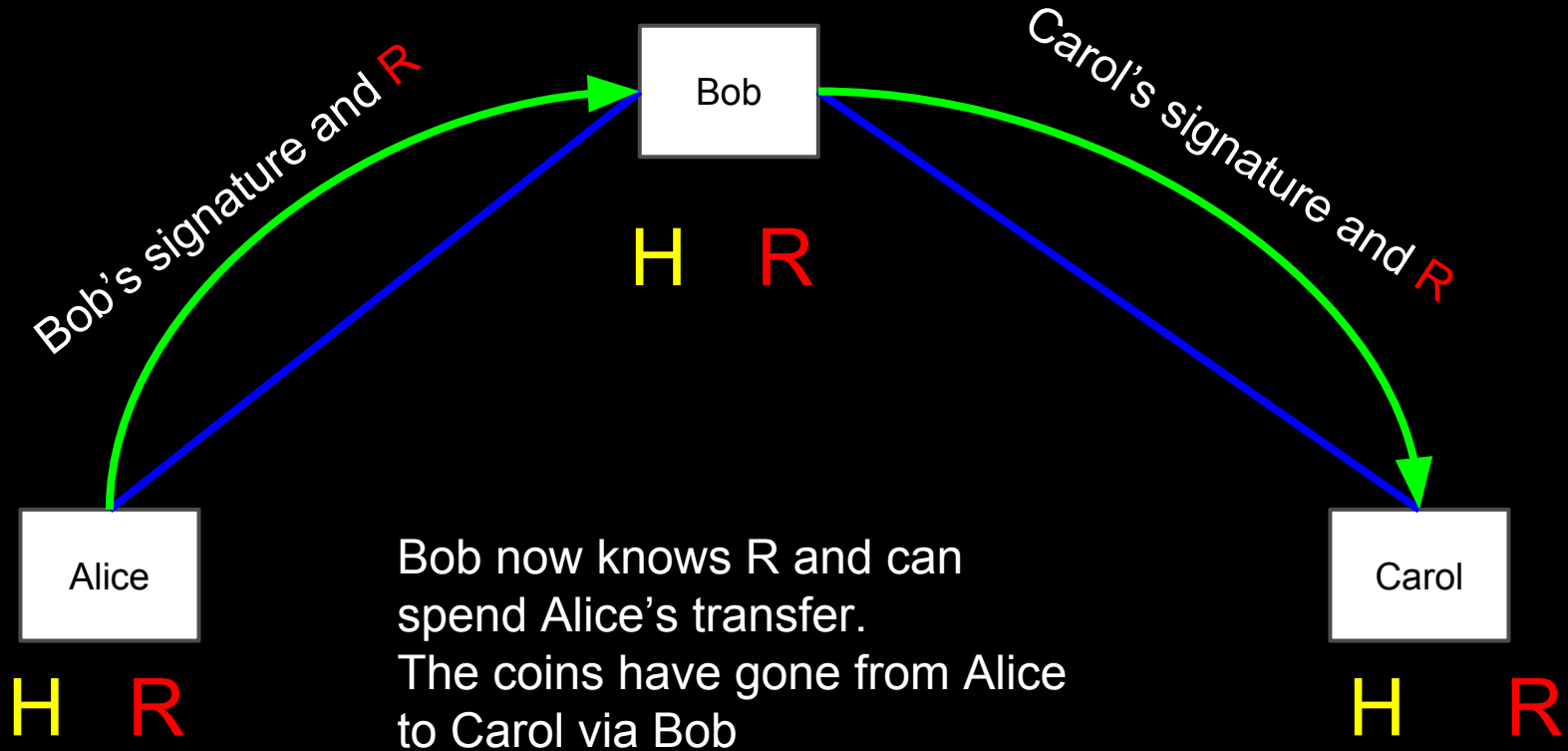
# Hash-Locked Contracts



# Hash-Locked Contracts



# Hash-Locked Contracts



# Problem!

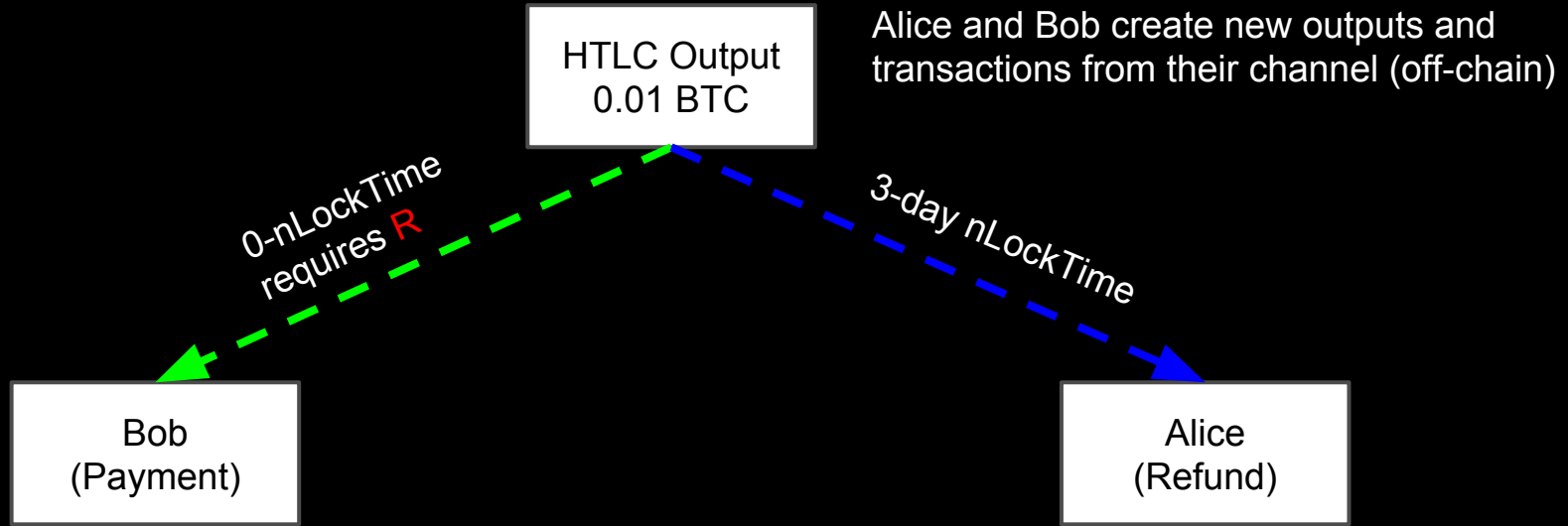
- If Carol refuses to disclose **R**, she will hold up the channel between Alice and Bob
    - If her channel expires after Alice and Bob's she can steal funds by redeeming the hashlock!
  - Bob has to be rich for this to really work
- 3rd party low-trust multisig and/or extremely small values sent can mostly work today

# Hashed Time-Lock Contract

1. If Bob can produce to Alice input R from hash H within 3 days, Alice will pay Bob 0.01 BTC
2. The above clause is void after 3 days
3. Either party may agree to settle terms using other methods if both agree
4. Violation of terms incurs a maximum penalty of funds in this contract



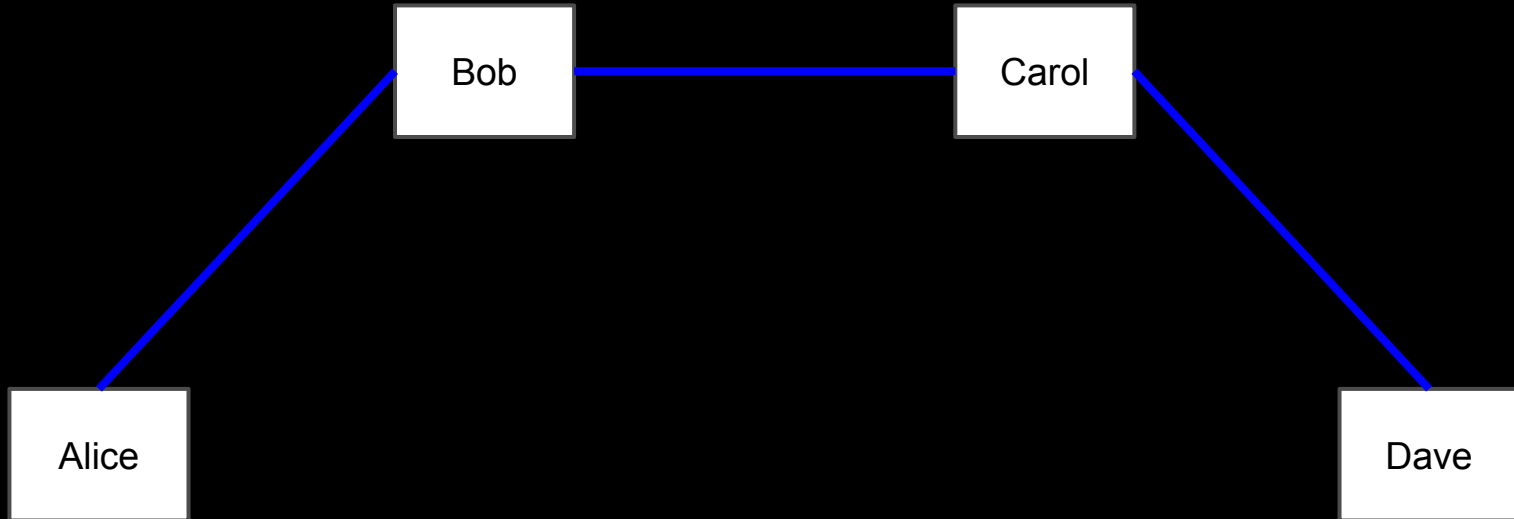
# Hashed Time-Lock Contract



```
OP_DEPTH 3 OP_EQUAL OP_IF OP_HASH160 <R> OP_EQUALVERIFY OP_0 2
<AlicePubkey1> <BobPubkey1> 2 OP_CHECKMULTISIG OP_ELSE OP_0 2
<AlicePubkey2> <BobPubkey2> 2 OP_CHECKMULTISIG OP_END
```

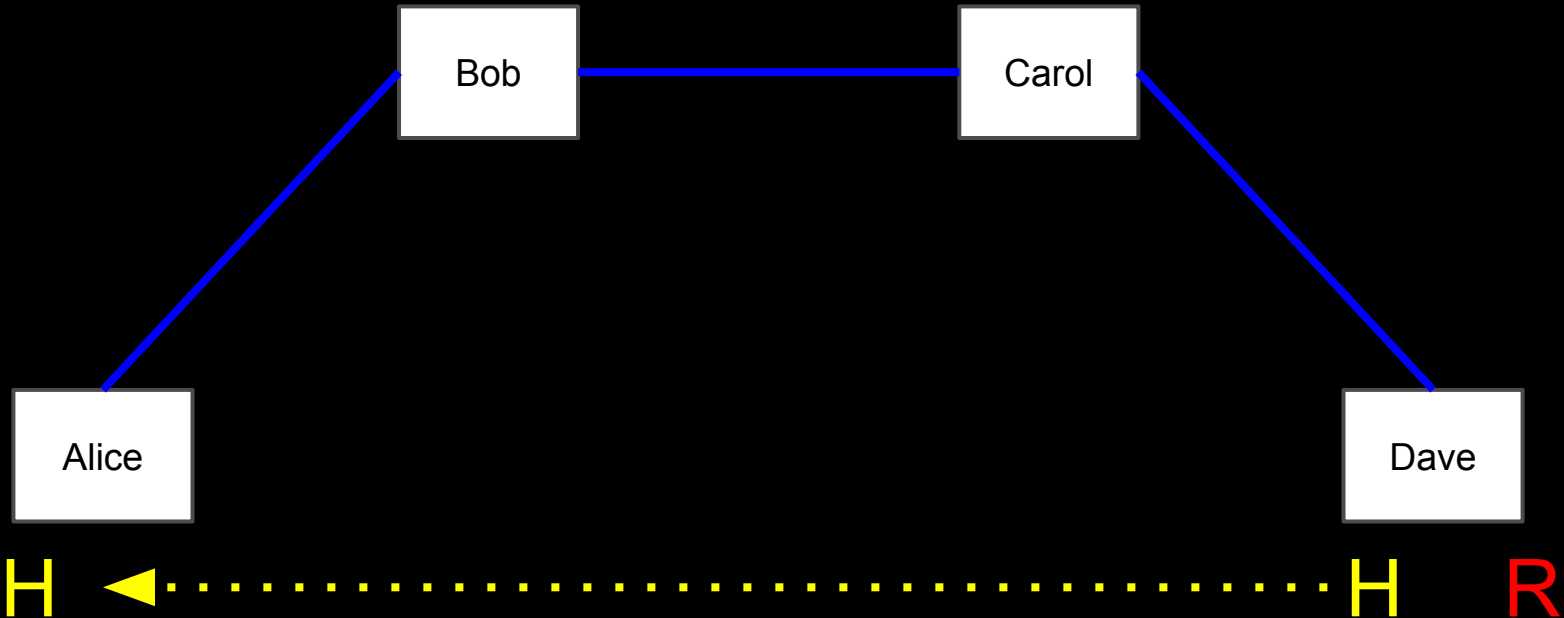
# Bitcoin Lightning Network

Alice wants to send funds to Dave via Bob and Carol

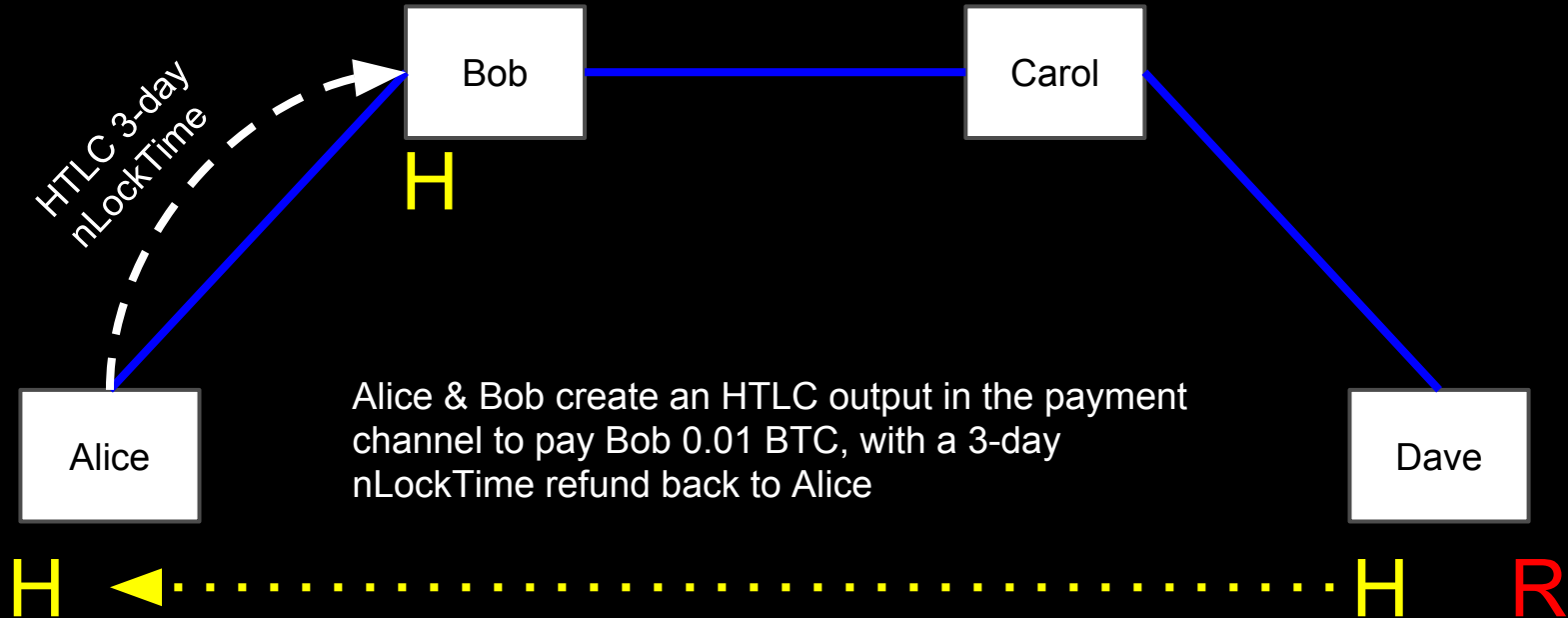


# Bitcoin Lightning Network

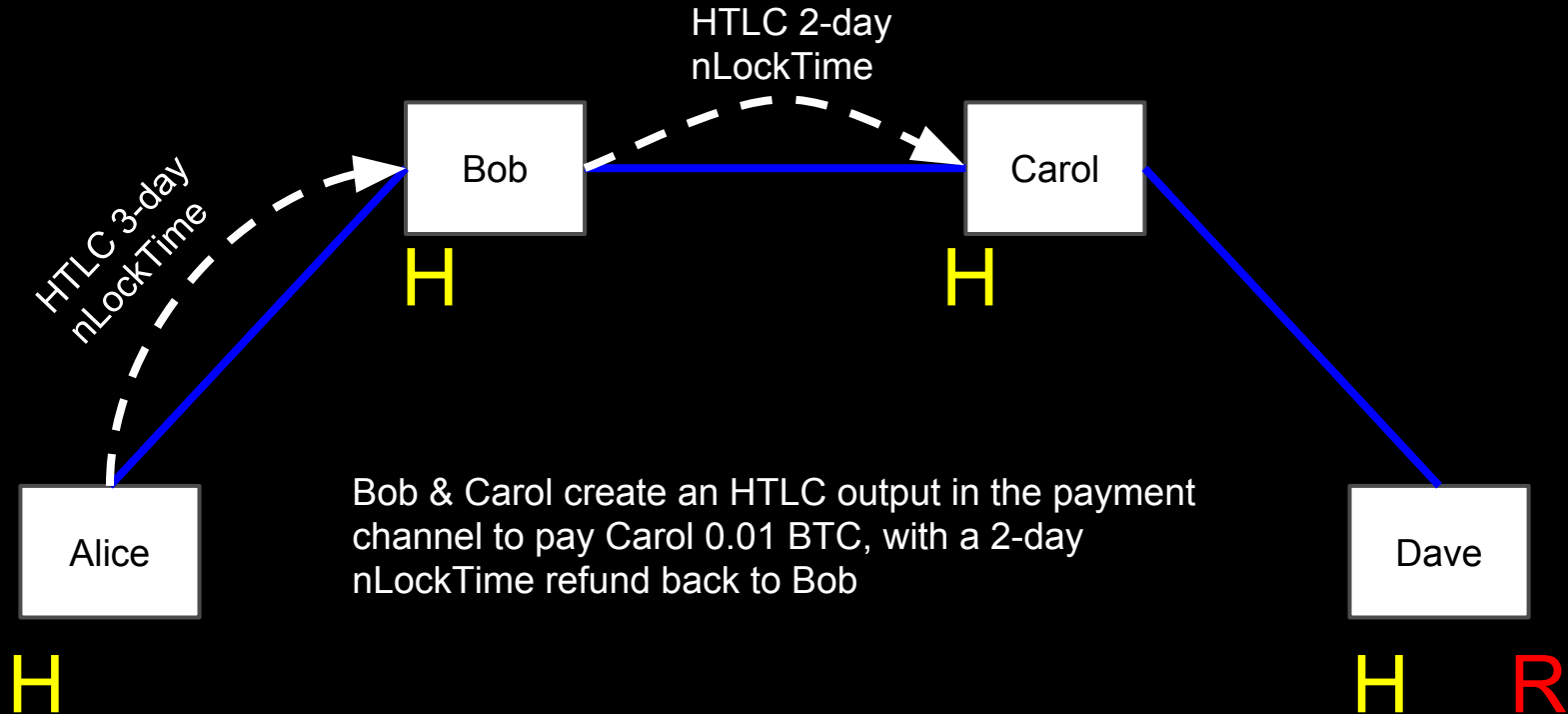
Dave sends Alice hash **H** produced from random data **R**



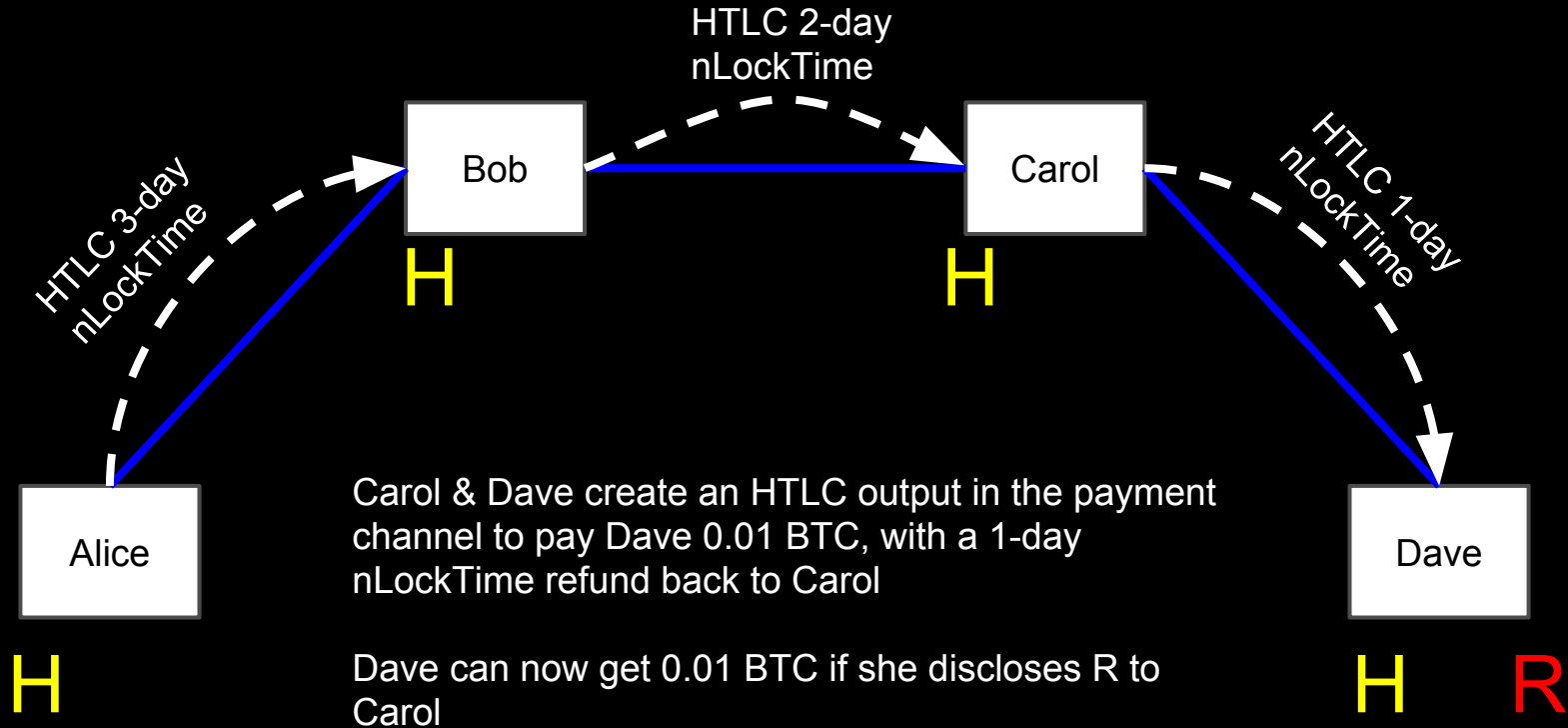
# Bitcoin Lightning Network



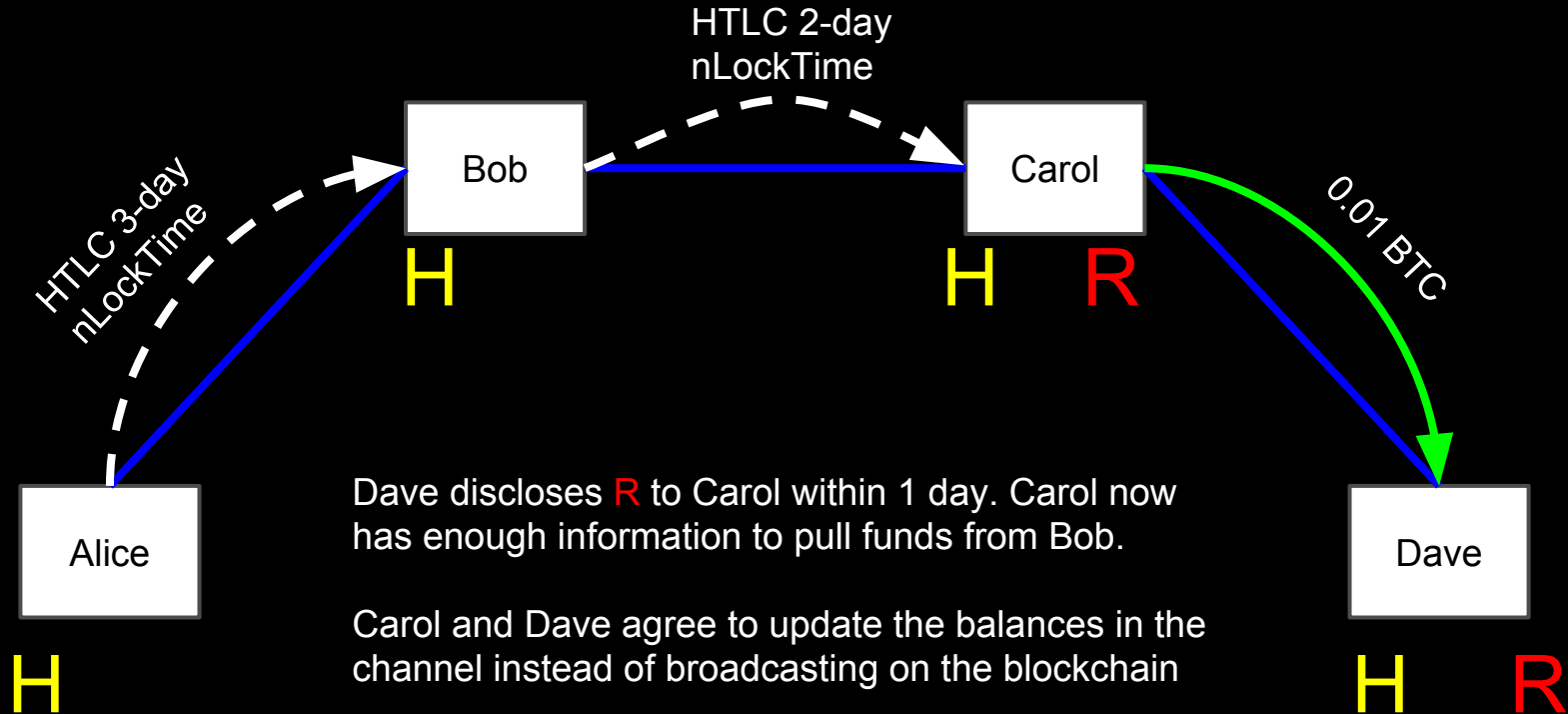
# Bitcoin Lightning Network



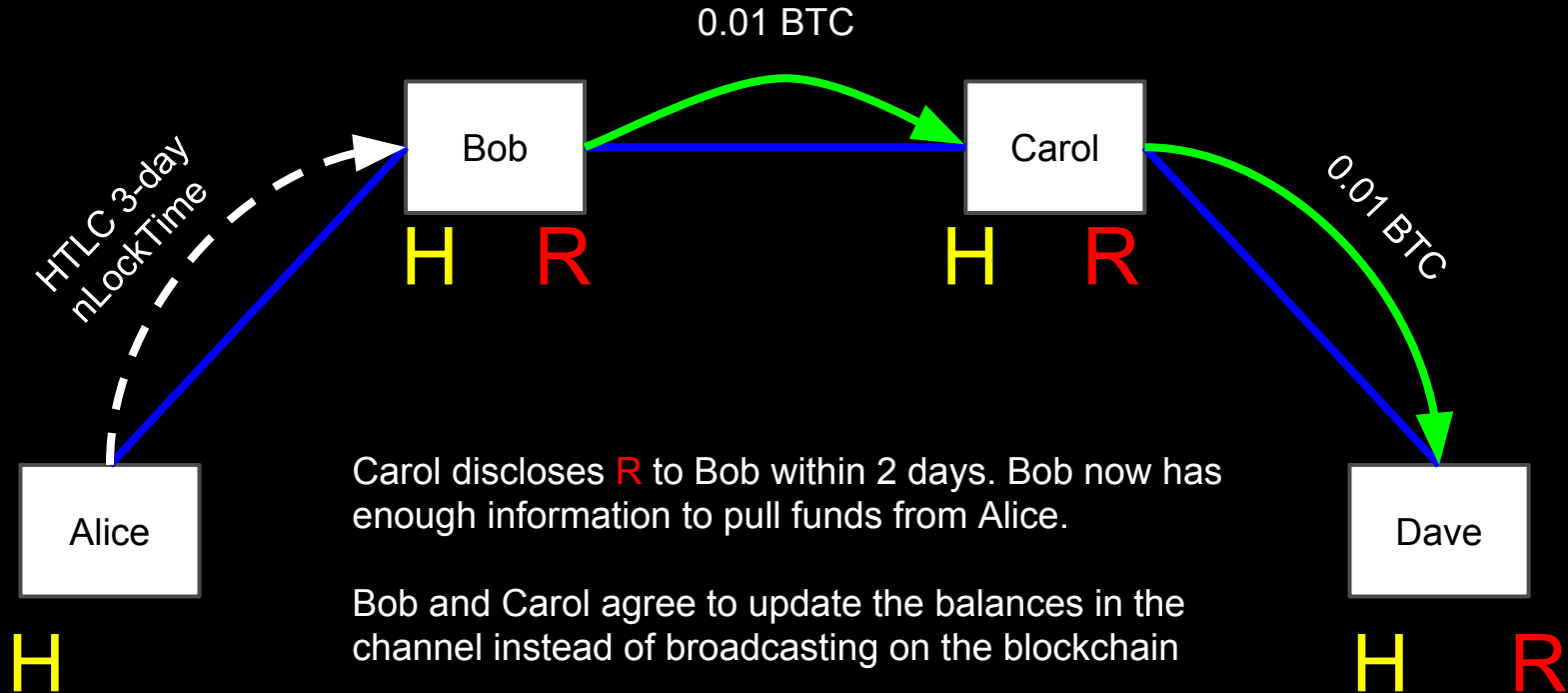
# Bitcoin Lightning Network



# Bitcoin Lightning Network

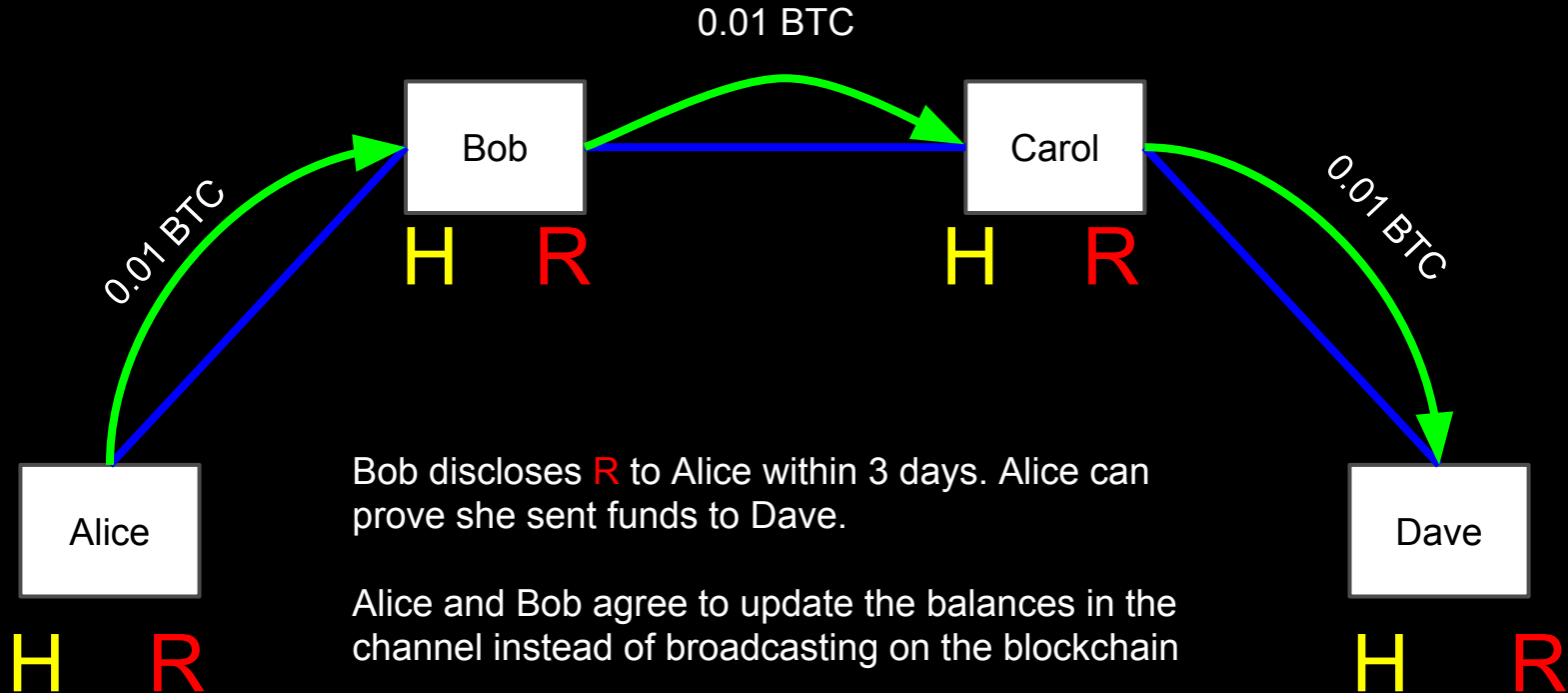


# Bitcoin Lightning Network

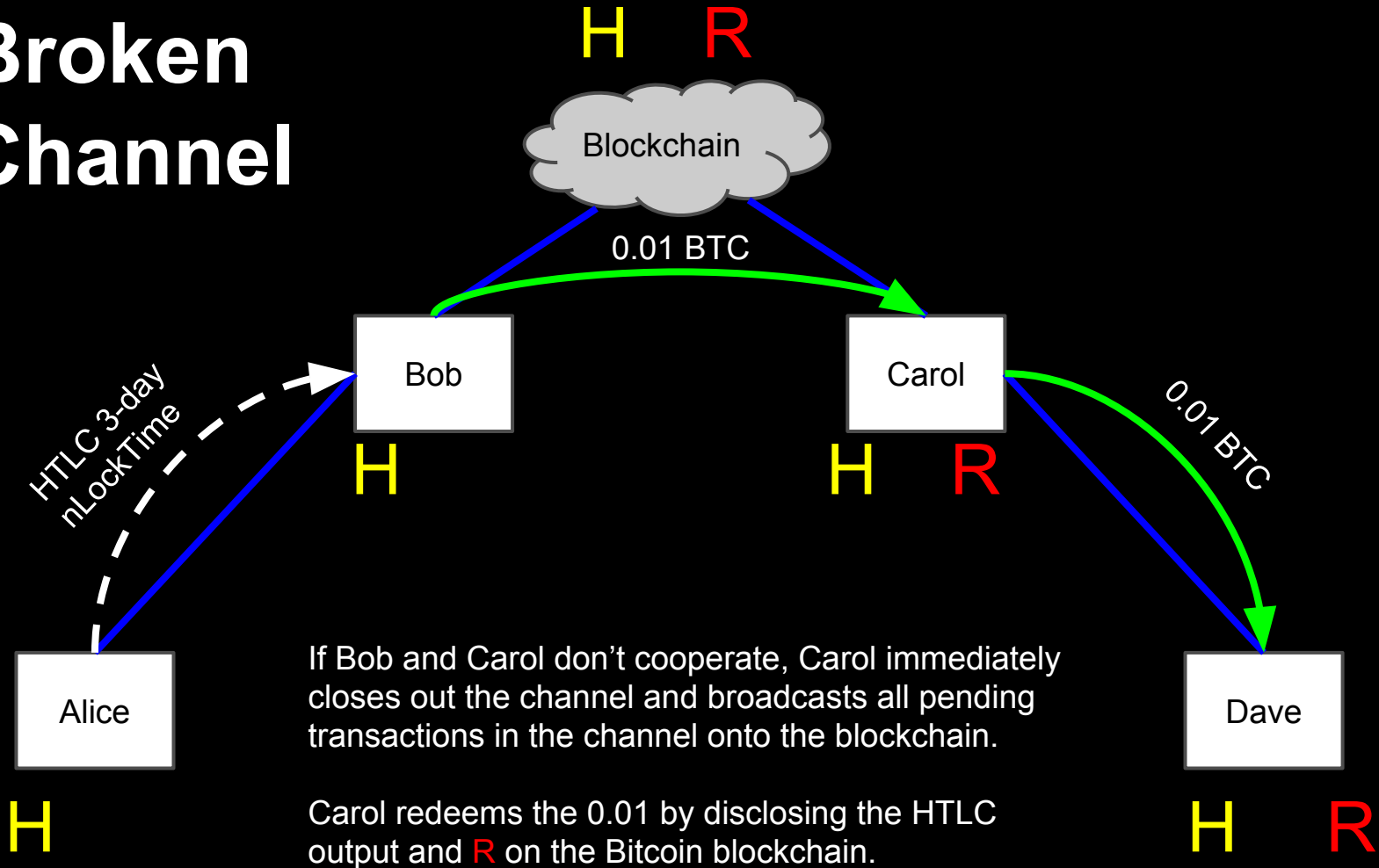




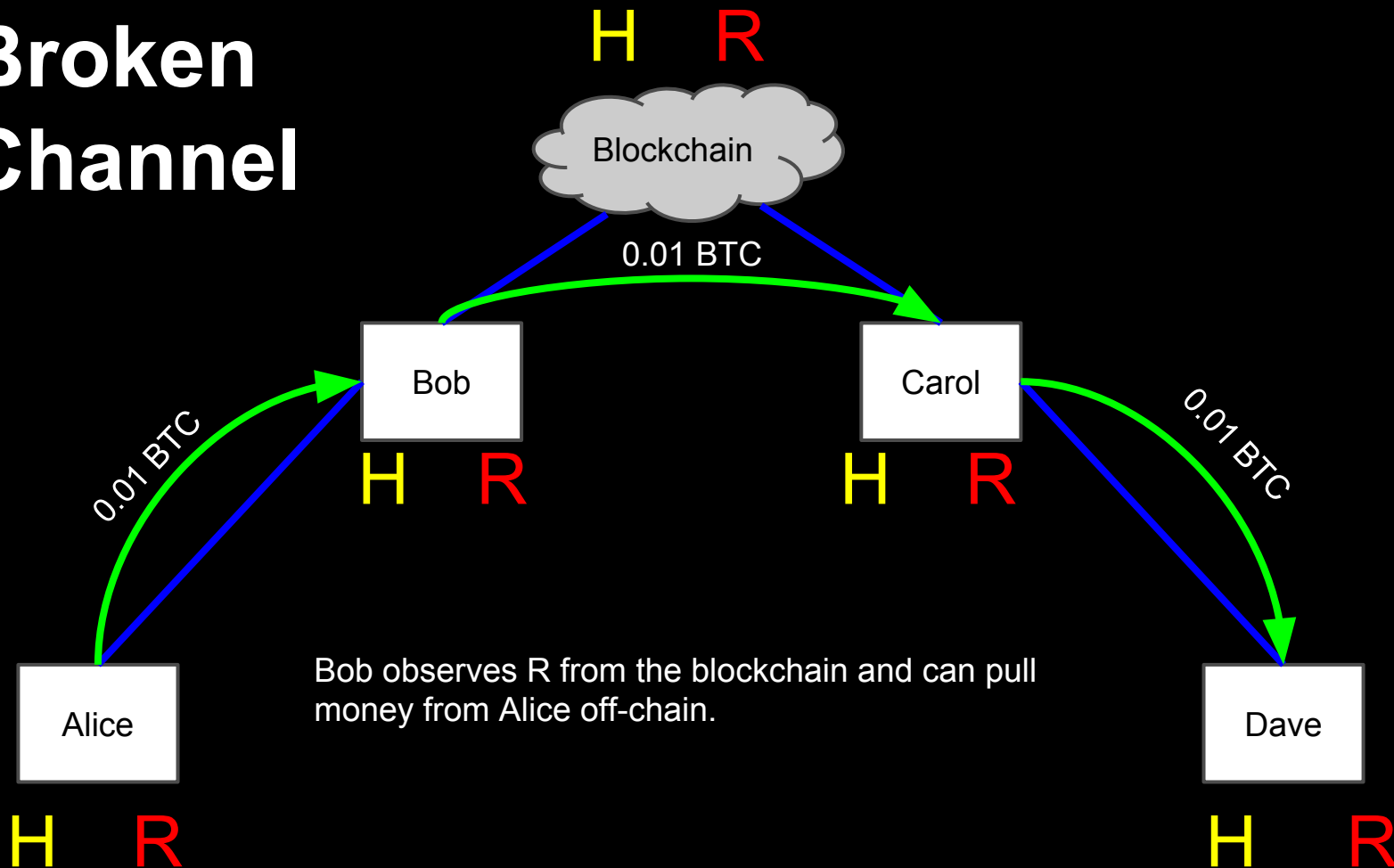
# Bitcoin Lightning Network



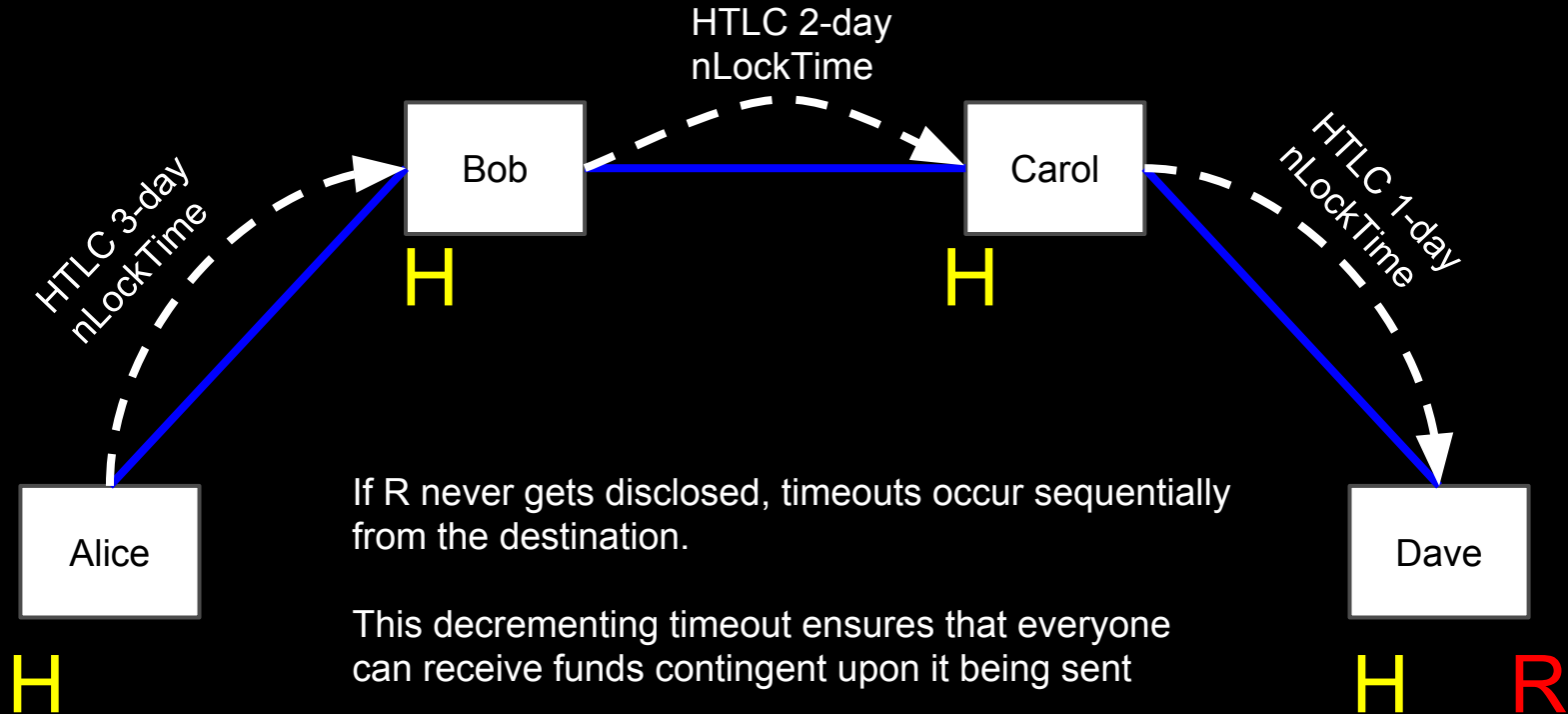
# Broken Channel



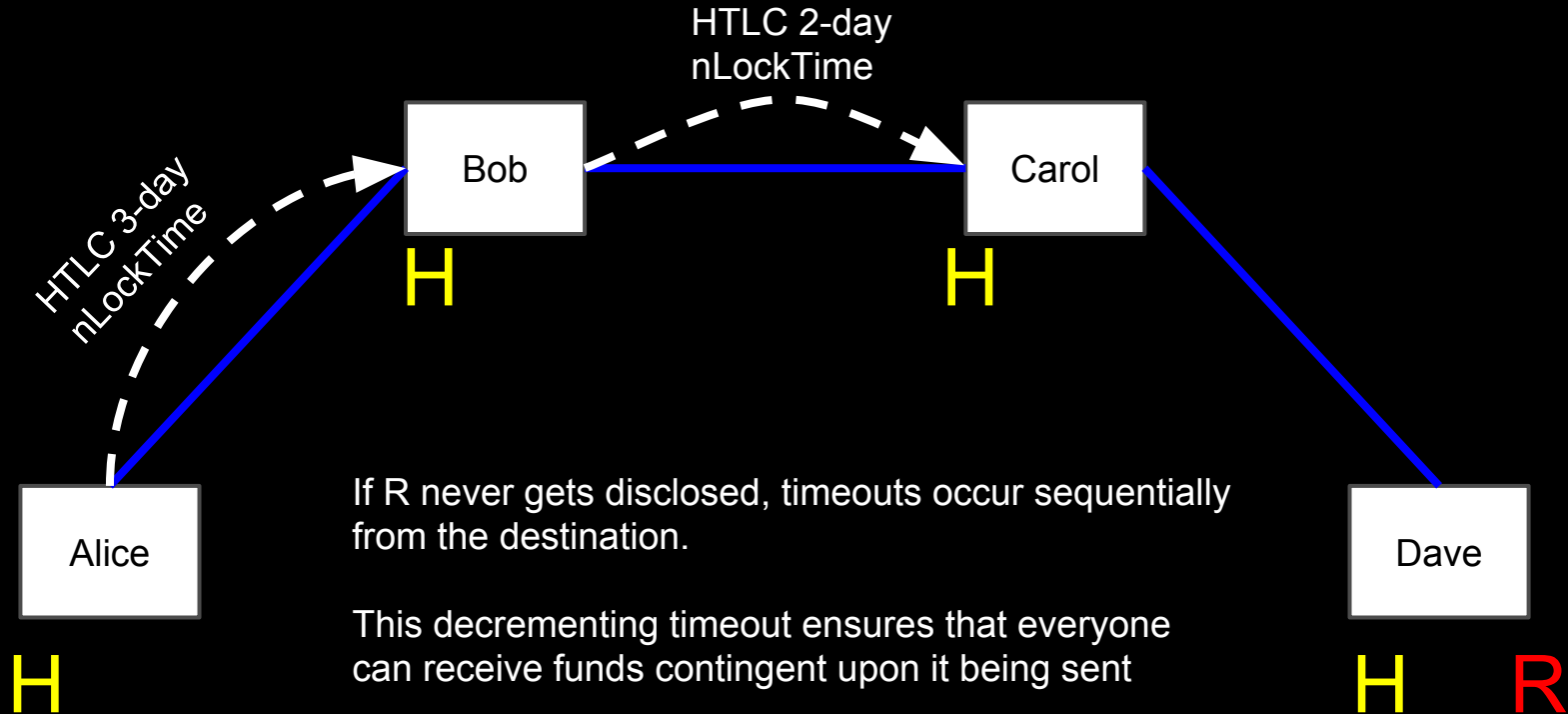
# Broken Channel



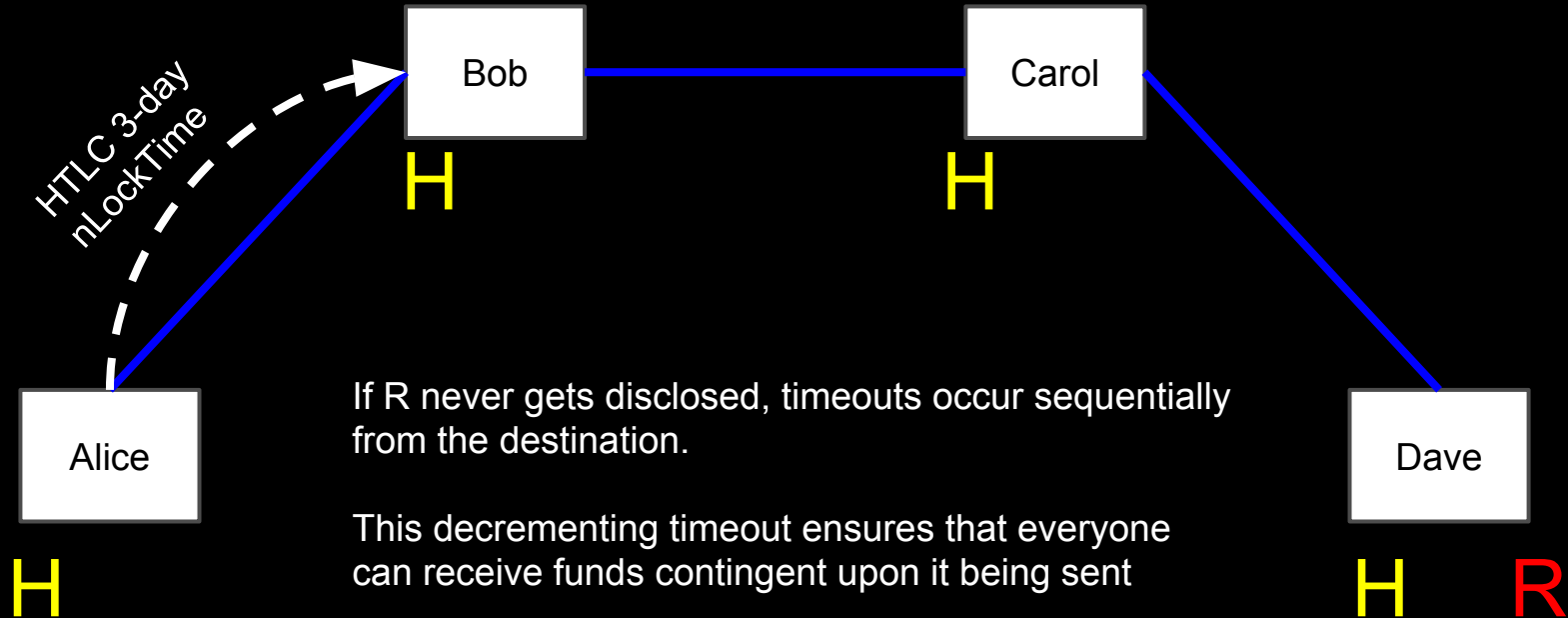
# Timeout



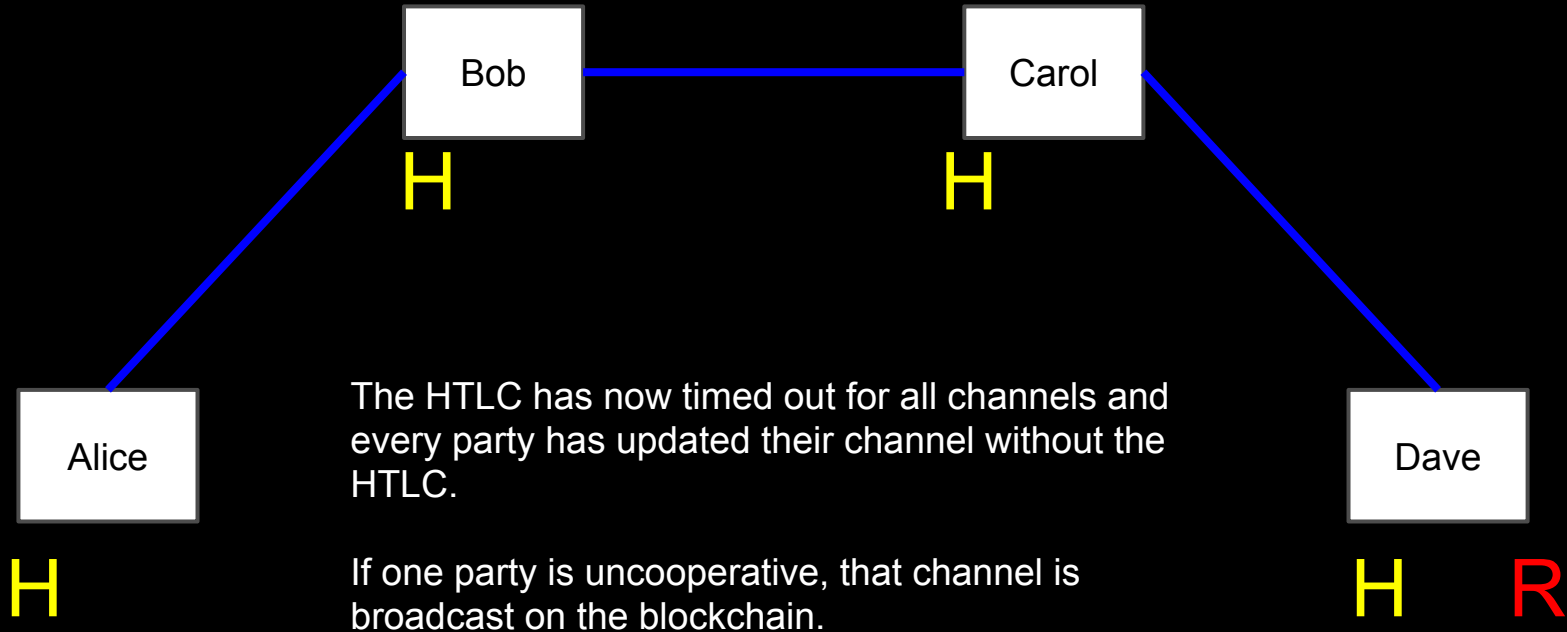
# Timeout



# Timeout



# Timeout



# Implications

- Blockchain transactions can be circuits instead of packets
  - Only uncooperative channels get broadcast early on the blockchain
  - Nearly all transactions occur off-chain securely with near-zero custodial default risk
  - Creating new channels are infrequent
- Relative OP\_CHECKLOCKTIMEVERIFY
- Instant Transactions, Scalable Micropayments



# Bitcoin Can Scale

- 7 billion people making 2 blockchain transactions per **day**
  - 24 GB blocks (~2,400 MB)
    - $7e9 * 2txs * 250Bytes / 144 \text{ block/day}$
  - ~50 Mbit/s best-case with IBLT @ 2tx/day/person
- 7 billion people roll-over 2 channels per year
  - 133 MB blocks - unlimited transactions count
    - $7e9 * 2 \text{ txs} * 500Byte / 52560 \text{ blocks/yr}$
  - ~3 Mbit/s with IBLT

# Bitcoin Can Scale

- Full Blockchain Archive
  - 7 TB/year (7,000 GB) /w ~unlimited transactions
  - \$300/year blockchain archival storage (2015 prices)
- Blockchain Pruning
  - Server nodes and miners only keeping recent blocks and UTXOs have enough storage today (2 TB, common in current-gen desktops)
- Mobile Wallets (Lightweight SPV clients)
  - Several megabytes of storage

