CS 168: Blockchain and Cryptocurrencies

Bitcoin Mining and UTXOs

Prof. Tom Austin
San José State University

Lab Review

Digital Currency – Ledger

Alice: 20
Bob: 11
Charlie: 5
David: 34

Alice

Bob

Alice: 20
Bob: 11
Charlie: 5
David: 34

"I am giving 10 cryptocoins to Bob"

David

Alice: 20
Bob: 11
Charlie: 5
David: 34

Charlie

Alice: 20
Bob: 11
Charlie: 5
David: 34

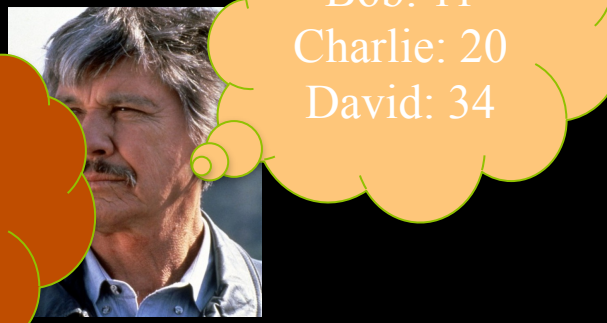
Digital Currency – Ledger

Alice

"I am giving 15 cryptocoins to Charlie"

David

Bob



Charlie

Digital Currency – Ledger

Alice

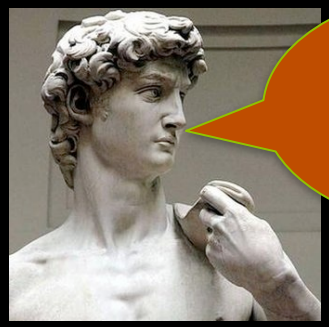
"I am giving 8 cryptocoins to David"

Bob

Invalid transaction!

Charlie

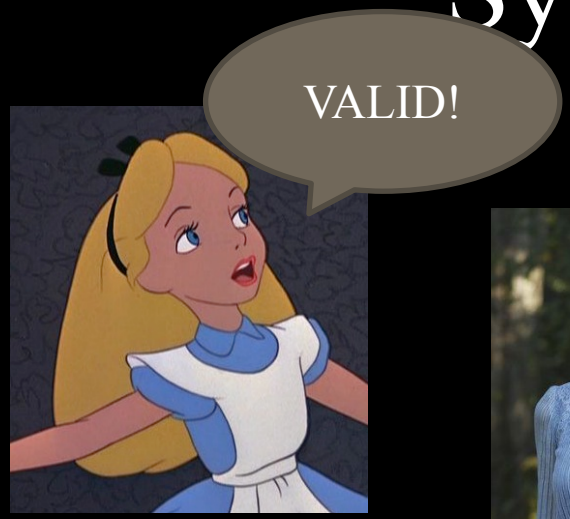
Invalid transaction!



David

Invalid transaction!

Sybil attack

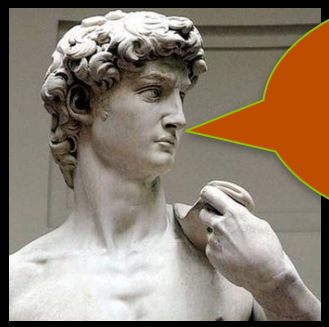


Alice

Alice 3

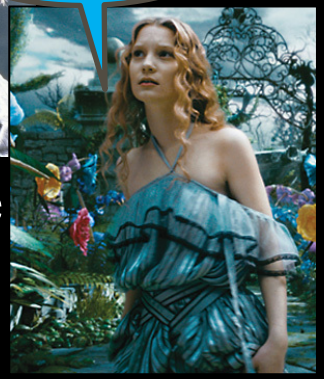
Bob

Alice 2



David

Charlie



Alice 4

How can we defend against Sybils?

- Know your clients, OR
- Force them to commit some resource
  - Proof-of-work
  - Proof-of-stake
  - Proof-of-storage
  - Others?

How can we defend against Sybils?

- Know your clients, OR
- Force them to commit some resource
  - ***Proof-of-work***
  - Proof-of-stake
  - Proof-of-storage
  - Others?


Bitcoin Miners

Nakamoto Consensus

- *Probabilistic*
- One CPU, one vote (Hah!)
- Open membership
- Solving PoW puzzle determines "leader"
  - i.e., who gets to make a block.
- More profitable to mine than to cheat

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

5		3		2	9	4						1			6		8			9						6	9	5		3		8
	2				1	7		5					7						2					7		3	2				9	
1					5			2				8		5		4		6		7				8			3					5
				8										7	8		4	9										7				
8			9		2		5					5		1	9		3	8		6					6		1		8			9
2	4	6		5	3			8																9			4	2		6	3	7
9	7									1					1		2					2									8	2
				3											5		6											1				
	5	8			6																						8			7	1	
										7	3	9								6	9	8										
						2			6	5												6	4			8						
									9		1		6	3				9	5		1		7									
3		5		1					1						6		1						1					1		3		9
	6										5				2		9				2										1	
		9	2	3							6					5					8							3	9	6		
8			1	6		3	4					4	5						6	8					3	5		8	2			4
		6												2				9												1		
4			9	8		5	2					8	6						5	7					6	8		7	4			5
		1	3	7							7					2					2							9	5	7		
	2										9				5		6				5										9	
9		4		2					2						1		3						1					2		4		3
									7		6		5	1				1	3		7		6									
						7			8	4												3	4			7						
										2	3	4								9	8	1										
	6	8			9																						6			3	7	
				7											2		8											1				
7	1									3					3		9					9									5	1
6	2	5		3	4			8																1			2	7		9	3	5
4			8		1		5					4		6	7		3	1		8					3		9		8			4
				5										3	6		1	2										4				
8					2			3				3		5		7		4		9				3			4					2
	9				3	6		5					4						3					2		5	7				4	
2		6		9	5	8						1			5		4			6						9	8	3		5		6

Bitcoin's Proof-of-Work

- Hashcash protocol
  - Uses cryptographic hashes
  - Designed for reducing email spam
- Target: number of (binary) leading zeroes required for hash value
- Nonce: random "number used once"

Proof-of-Work Mining

- Process:
  - "Miners" choose a nonce
  - If `hash(block)` meets target, share block
  - Otherwise, continue searching
- N = number of leading zeroes required for target
- Work to find valid proof:
  - 2^N hashes
- Work to verify proof:
  - 1 hash

Sample Block

Blockchain.com Wallet Exchange Explorer Buy Bitcoin Trade

Sponsored Content

Explorer > Bitcoin Explorer > Block

Search your transaction, an address or a block

USD

Block 689933

USD BTC

This block was mined on July 06, 2021 at 12:18 PM PDT by AntPool. It currently has 1 confirmations on the Bitcoin blockchain.

The miner(s) of this block earned a total reward of 6.25000000 BTC ($211,708.31). The reward consisted of a base reward of 6.25000000 BTC ($211,708.31) with an additional 0.48555617 BTC ($16,447.40) reward paid as fees of the 2065 transactions which were included in the block. The Block rewards, also known as the Coinbase reward, were sent to this address.

A total of 16,558.94772863 BTC ($560,906,700.86) were sent in the block with the average transaction being 8.01886089 BTC ($271,625.52). Learn more about how blocks work.

Hash	0000000000000000000063b7a1703e5b2e373f503895774d7ec90881ab9793d1f
Confirmations	1
Timestamp	2021-07-06 12:18
Height	689933
Miner	AntPool
Number of Transactions	2,065
Difficulty	14,363,025,673,659,97

Mining

- Miners hash transaction details plus a "proof"
  - Reward: newly generated bitcoins (*coinbase transaction*)
- Cost to discover proof
  - 2^N hashes
- Cost to verify proof
  - 1 hash

Digital Currency – Proof of Work

Alice

"I am giving 2 cryptocoins to Charlie"

Bob

David

Charlie

Digital Currency – Proof of Work

Alice: 3+1
Bob: 11
Charlie: 22
David: 34

Searching for proof of work…

Alice

"I am giving 2 cryptocoins to Charlie"

Alice: 3
Bob: 11+1
Charlie: 22
David: 34

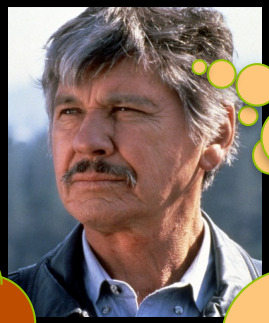
Bob

Searching for proof of work…

Alice: 3
Bob: 11
Charlie: 22
David: 34 +1

David

Searching for proof of work…



Charli

Searching for proof of work…

Alice: 3
Bob: 11
Charlie: 22+1
David: 34

Digital Currency – Proof of Work

Alice

Bob

Found proof!

Charlie

"I am giving 2 cryptocoins to Charlie"

David

Digital Currency – Proof of Work

Alice: 3
Bob: 11
Charlie: 22
David: 35

Alice

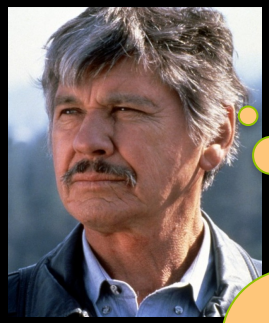
"I am giving 2 cryptocoins to Charlie"

Bob

Alice: 3
Bob: 11
Charlie: 22
David: 35

Alice: 3
Bob: 11
Charlie: 22
David: 35

David



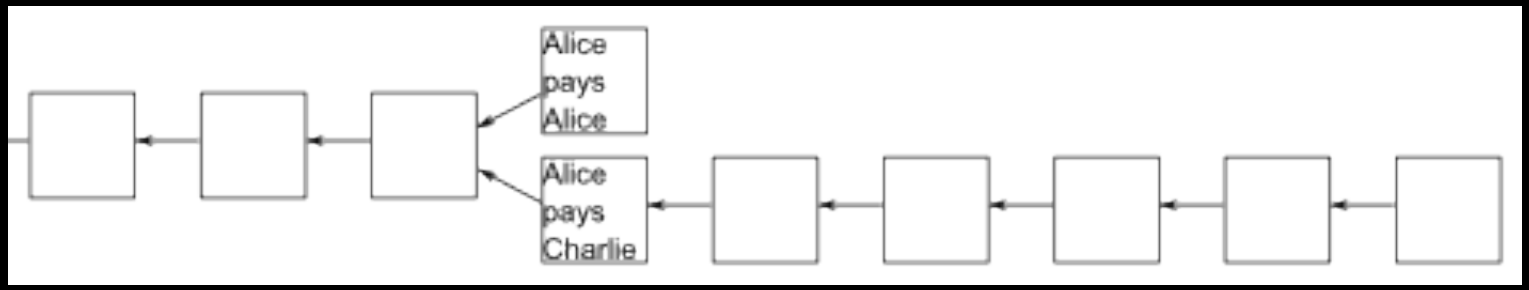
Charli

Alice: 3
Bob: 11
Charlie: 22
David: 35

Handling Discrepancies

Chains **weighted by work** used in their creation.

Generally, this means the longest blockchain.



Review BlockExplorer
(in class)

Available at https://btc.com/

How many leading zeroes?

```
0000000000000000000004cf5d4bcee3bec
16c40cc823fb1aeb75c5d9384797c82
```

Lab: Proof-of-work

In today's lab, we will review the hash-cash algorithm used for proof-of-work (PoW) in Bitcoin.

```
utils.hash(s+proof)
```

See Canvas for details.

UTXO Model

Two Models of Tracking Balances

- Account-based model
  - Track balances for each account
  - Simpler
  - Used by SpartanGold
- UTXO-based model
  - Track unspent coins
  - More complex, but (slightly?) better pseudo-anonymity
  - Used by Bitcoin

What does “UTXO” stand for?

U – Unspent

TX – Transaction

O – Output

Transaction Chains

- Not the same as blockchains
- Each transaction output can be a future transaction input.
- Each output can only be spent once
- To know what bitcoins are available you only need to keep track of the **Unspent Transaction Outputs (UTXOs)**.

Double-entry Bookkeeping

- Each transaction specifies *inputs* and *outputs*
- **All inputs must be spent**
  - Transaction fee = sum(inputs) – sum(outputs)
  - Change address = Address spender gives to reclaim unused bitcoins.
- Special case: coinbase transactions
  - New coins generated as a reward for miners.

Transaction Chains

Figure from *Mastering Bitcoin*

Transaction Chains

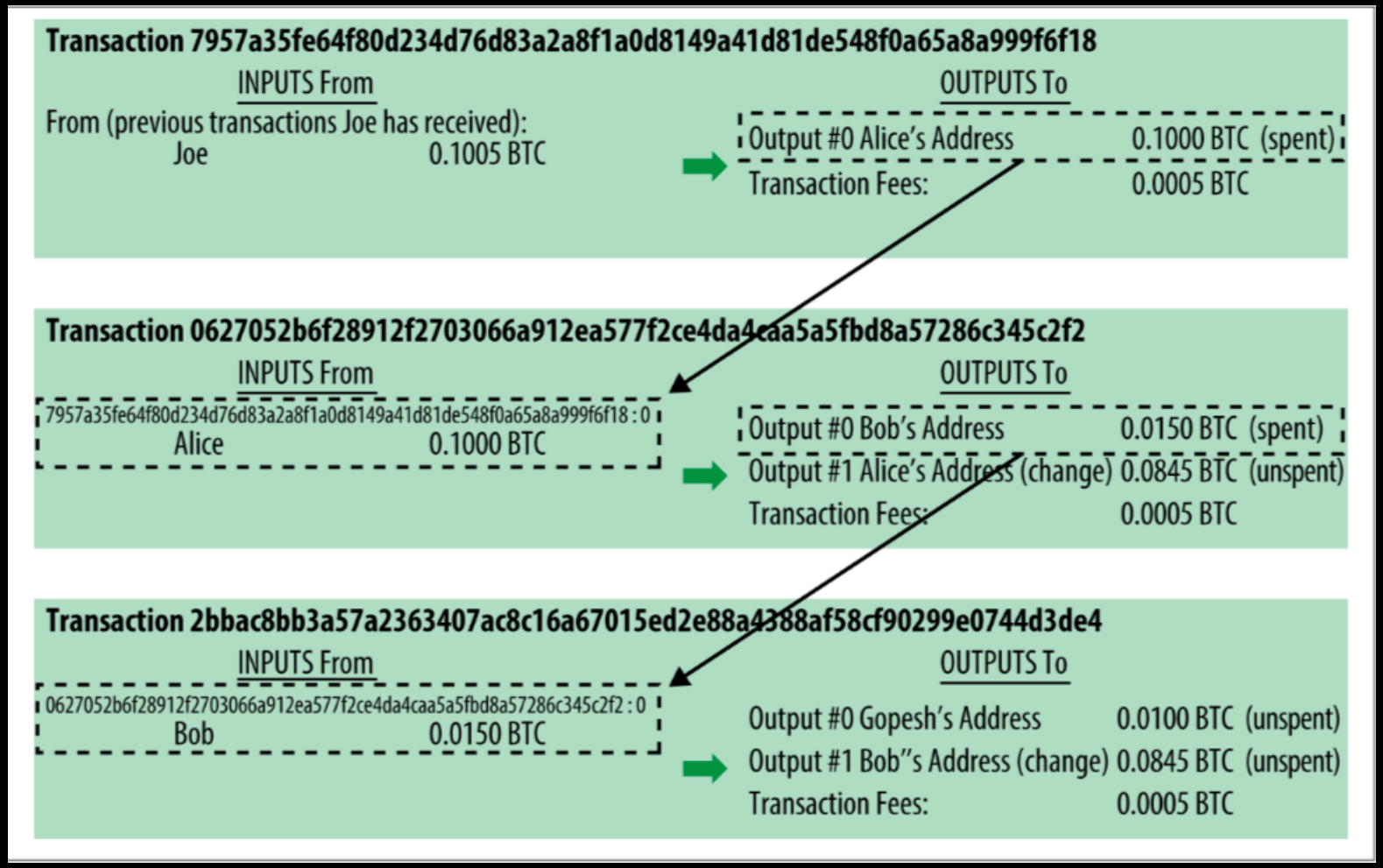


Figure from *Mastering Bitcoin*

Transaction Forms

Common Transactions

Most typical: Alice pays Bob, and keeps the change

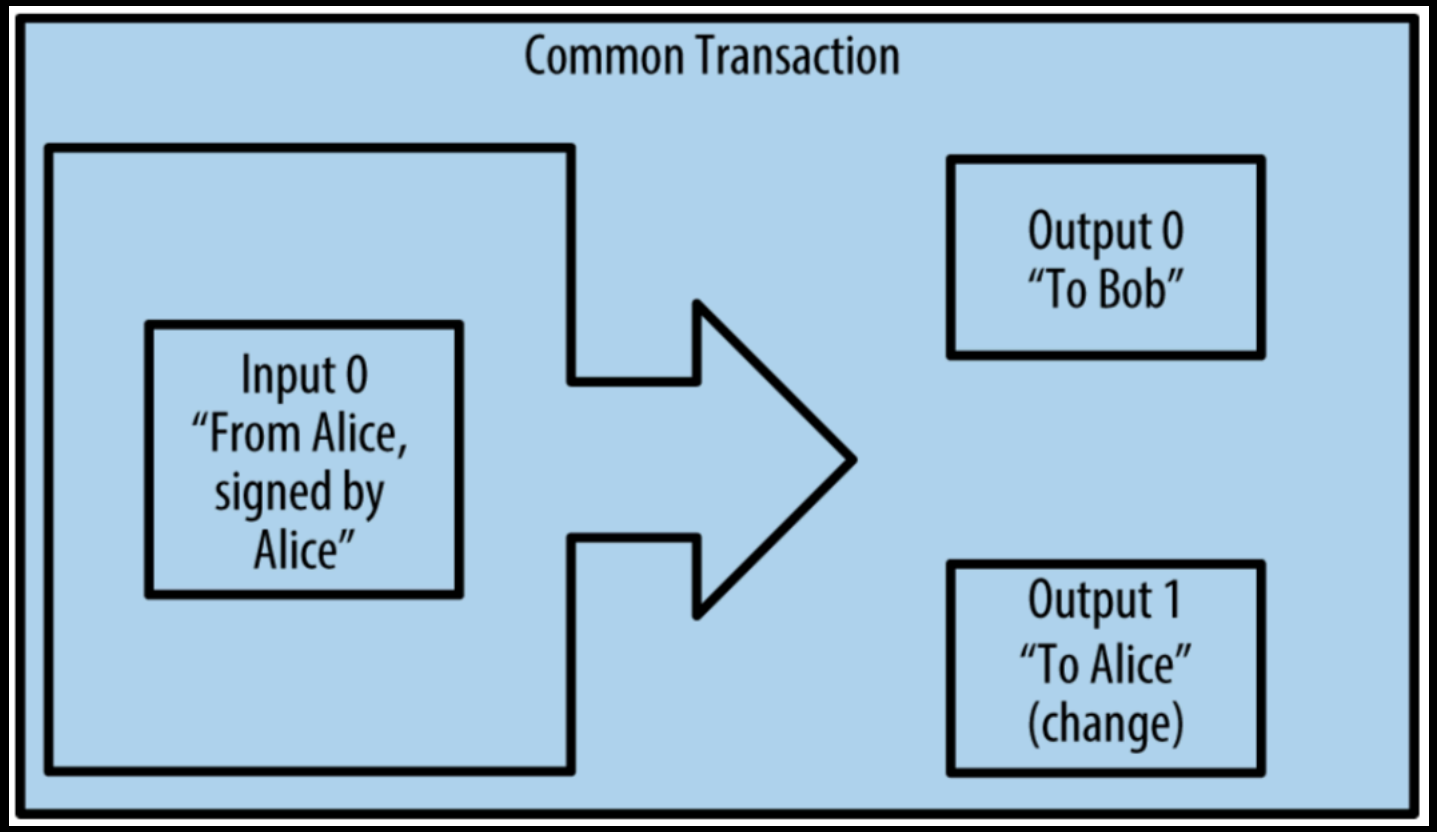


Figure from *Mastering Bitcoin*

Aggregating Transaction

Alice has many private keys and wants to combine them.

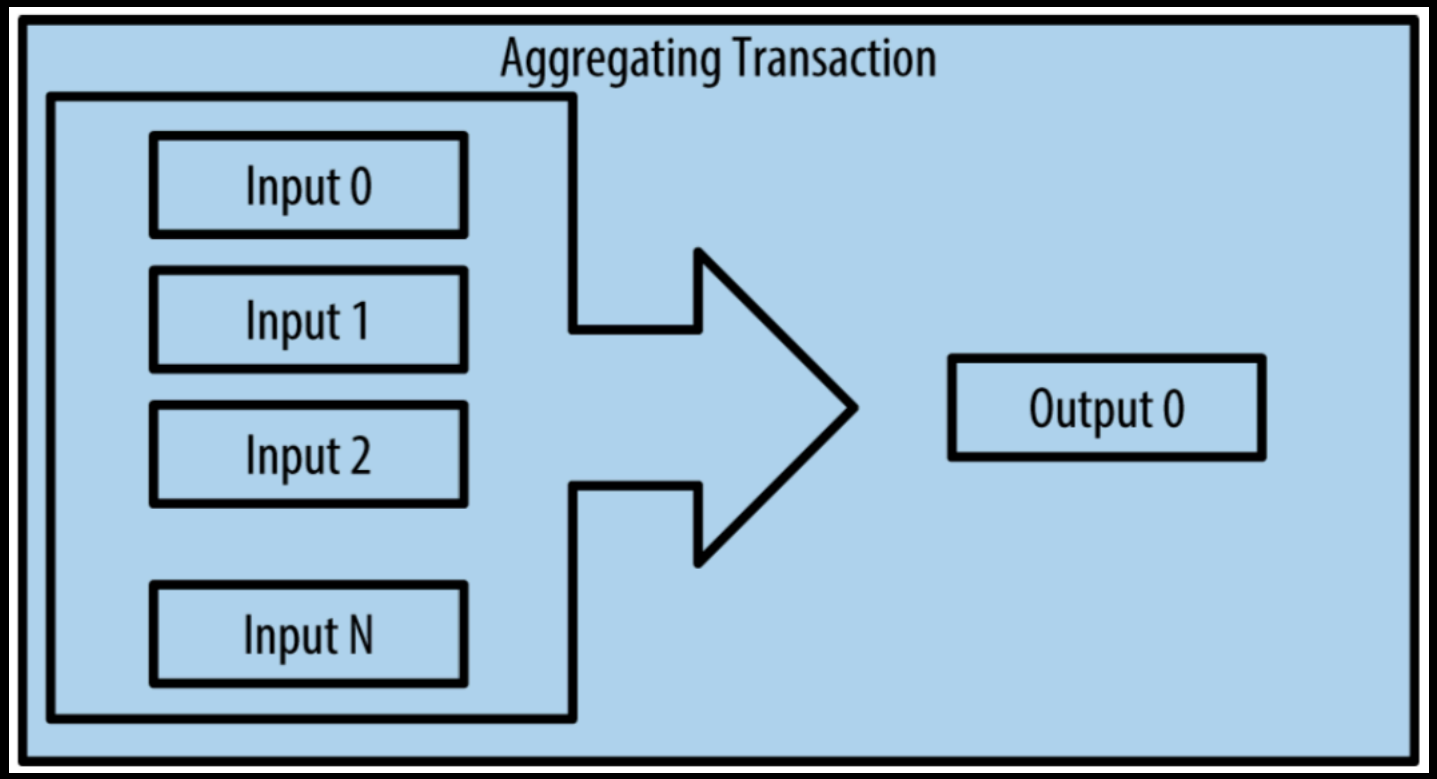


Figure from *Mastering Bitcoin*

Distributing Transaction

Alice pays several different people simultaneously.

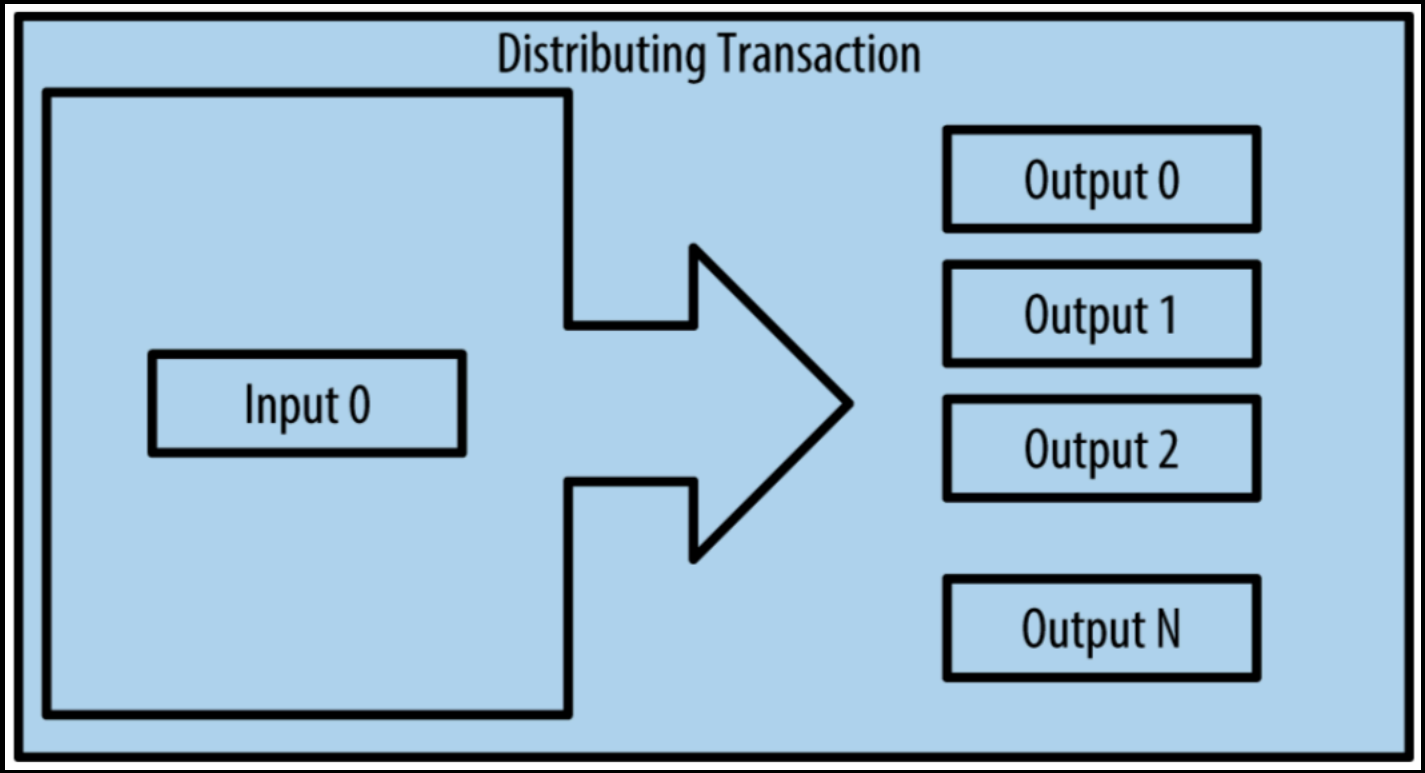


Figure from *Mastering Bitcoin*

HW2: Adding UTXO model to SpartanGold

- Main change: multiple inputs allowed
- The `from`, `pubKey`, and `sig` fields:
  - Single values in standard SpartanGold
  - Need to be arrays for the UTXO model
- Some differences from Bitcoin
  - We will make transaction fee explicit

Standard JSON for sample transaction
(parts elided with "…")

```
{ from: '4HWTOR8cgvejeMd…',
  nonce: 0,
  pubKey: '-----BEGIN PUBLIC KEY-----\n' …,
  sig: '83adb439…',
  fee: 1,
  outputs: [
   { amount: 25, address: 'vAy8w7bavN9…' }
  ],
  data: {}
}
```

SpartanGold: Adding in Inputs


```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```

SpartanGold: Adding in Inputs

```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```

Outputs (addresses) from previous transactions

SpartanGold: Adding in Inputs


```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```


Not needed

SpartanGold: Adding in Inputs


```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```


Public keys matching previous outputs

SpartanGold: Adding in Inputs


```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```


Signatures for previous outputs

SpartanGold: Adding in Inputs

```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```

Address, key, and signature tying back to **first** previous output

SpartanGold: Adding in Inputs

```
{
  "from": [
    "I3ZtgbDVXVTutleSYHNQ2Hzt9GY5rr4mdRtOs1c+7xw=",
    "Y6AhPOu8NsGzfVVlcDjmPpMHkw0mYUKPaPQblMrdxvU="
  ],
  "nonce": 0,
  "pubKey": [
    "-----BEGIN PUBLIC KEY----- ...",
    "-----BEGIN PUBLIC KEY----- ..."
  ],
  "sig": [
    "8203f25f23bcc1f5281b1126fa555e2611...",
    "d8a0e624e4972053dfbd9fd5d68d3512f5..."
  ],
  ...
```

Address, key, and signature tying back to **second** previous output

SpartanGold: Outputs


```
  ...,
  "fee": 1,
  "outputs": [
    {
      "amount": 110,
      "address":
        "j+9fRcMFTVX5hnNzwmL1U8QskUTGSwMehc/nhQGWM5k="
    },
    {
      "amount": 28,
      "address":
        "XVW5aEtvwVOUyo1+zjjnkFHmkOA1I2t/HBMwj/nQQT0="
    }
  ],
   "data": {}
}
```

SpartanGold: Outputs

Transaction fee

```
  ...,
  "fee": 1,
  "outputs": [
    {
      "amount": 110,
      "address":
        "j+9fRcMFTVX5hnNzwmL1U8QskUTGSwMehc/nhQGWM5k="
    },
    {
      "amount": 28,
      "address":
        "XVW5aEtvwVOUyo1+zjjnkFHmkOA1I2t/HBMwj/nQQT0="
    }
  ],
   "data": {}
}
```

SpartanGold: Outputs

```
  ...,
  "fee": 1,
  "outputs": [
    {
      "amount": 110,
      "address":
        "j+9fRcMFTVX5hnNzwmL1U8QskUTGSwMehc/nhQGWM5k="
    },
    {
      "amount": 28,
      "address":
        "XVW5aEtvwVOUyo1+zjjnkFHmkOA1I2t/HBMwj/nQQT0="
    }
  ],
    "data": {}
}
```

Outputs (addresses) after this round

SpartanGold: Outputs

```
  ...,
  "fee": 1,
  "outputs": [
    {
      "amount": 110,
      "address":
        "j+9fRcMFTVX5hnNzwmL1U8QskUTGSwMehc/nhQGWM5k="
    },
    {
      "amount": 28,
      "address":
        "XVW5aEtvwVOUyo1+zjjnkFHmkOA1I2t/HBMwj/nQQT0="
    }
  ],
   "data": {}
}
```

First UTXO

SpartanGold: Outputs

NOTE: If you have all UTXOs that have not yet been used as a transaction input, you know all available balances.

```
  ...,
  "fee": 1,
  "outputs": [
    {
      "amount": 110,
      "address":
        "j+9fRcMFTVX5hnNzwmL1U8QskUTGSwMehc/nhQGWM5k="
    },
    {
      "amount": 28,
      "address":
        "XVW5aEtvwVOUyo1+zjjnkFHmkOA1I2t/HBMwj/nQQT0="
    }
  ],
  "data": {}
}
```

Second UTXO

SpartanGold: Outputs

```
  ...,
  "fee": 1,
  "outputs": [
    {
      "amount": 110,
      "address":
        "j+9fRcMFTVX5hnNzwmL1U8QskUTGSwMehc/nhQGWM5k="
    },
    {
      "amount": 28,
      "address":
        "XVW5aEtvwVOUyo1+zjjnkFHmkOA1I2t/HBMwj/nQQT0="
    }
  ],
   "data": {}
}
```

Addresses used to reference UTXOs as future inputs

Showing transaction chains for SpartanGold

(in-class)

HW 2: Review starter code