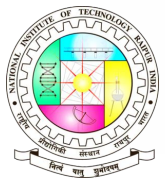
Bitcoin Transactions

Dr. Preeti Chandrakar
Assistant Professor

Department of Computer Science and Engineering
National Institute of Technology, Raipur
September 2021

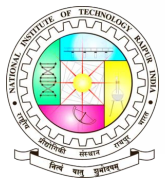
Outline

- Introduction
- Process of transaction
- Double-entry bookkeeping ledger
- Transaction Outputs and Inputs Transaction Outputs
- UTXO
- Transaction fee



Introduction

- A transaction is basically transfer of authorization of value from one to another owner
- In bitcoin, a transaction is transfer ownership of some bitcoin value from one owner to other and other to next
- Transactions are the most important part of the bitcoin system.
- Everything else in bitcoin is designed to ensure that transactions can be created, propagated on the network, validated
- And lastly added to the global ledger of transactions (the blockchain)
- Transactions are data structures that encode the transfer of value between participants in the bitcoin system.
- Each transaction is a public entry in bitcoin's blockchain, the global double-entry bookkeeping ledger.



Process of transaction

Yuvraj

I am authorized to spend these100 ฿

Transferring 80 ฿ to Shreya, signed with private key, digital signature

Shreya

I am authorized to spend these 79.8 ฿

Transferring 30 ฿ to Divya, signed with private key, digital signature

Divya

I am authorized to spend these 29.8 ฿

Transferring 30 ฿ to Anurag, signed with private key, digital signature

Anurag

I am authorized to spend these 29.8 ฿

*for each transaction 0.2 ฿ is deducted as transaction

Double-entry bookkeeping ledger

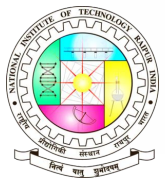
Transaction as Double-Entry Bookkeeping

Inputs	Value	Outputs	Value
Input 1	0.10 BTC	Output 1	0.10 BTC
Input 2	0.20 BTC	Output 2	0.20 BTC
Input 3	0.10 BTC	Output 3	0.20 BTC
Input 4	0.15 BTC		
Total Inputs:	0.55 BTC	Total Outputs:	0.50 BTC

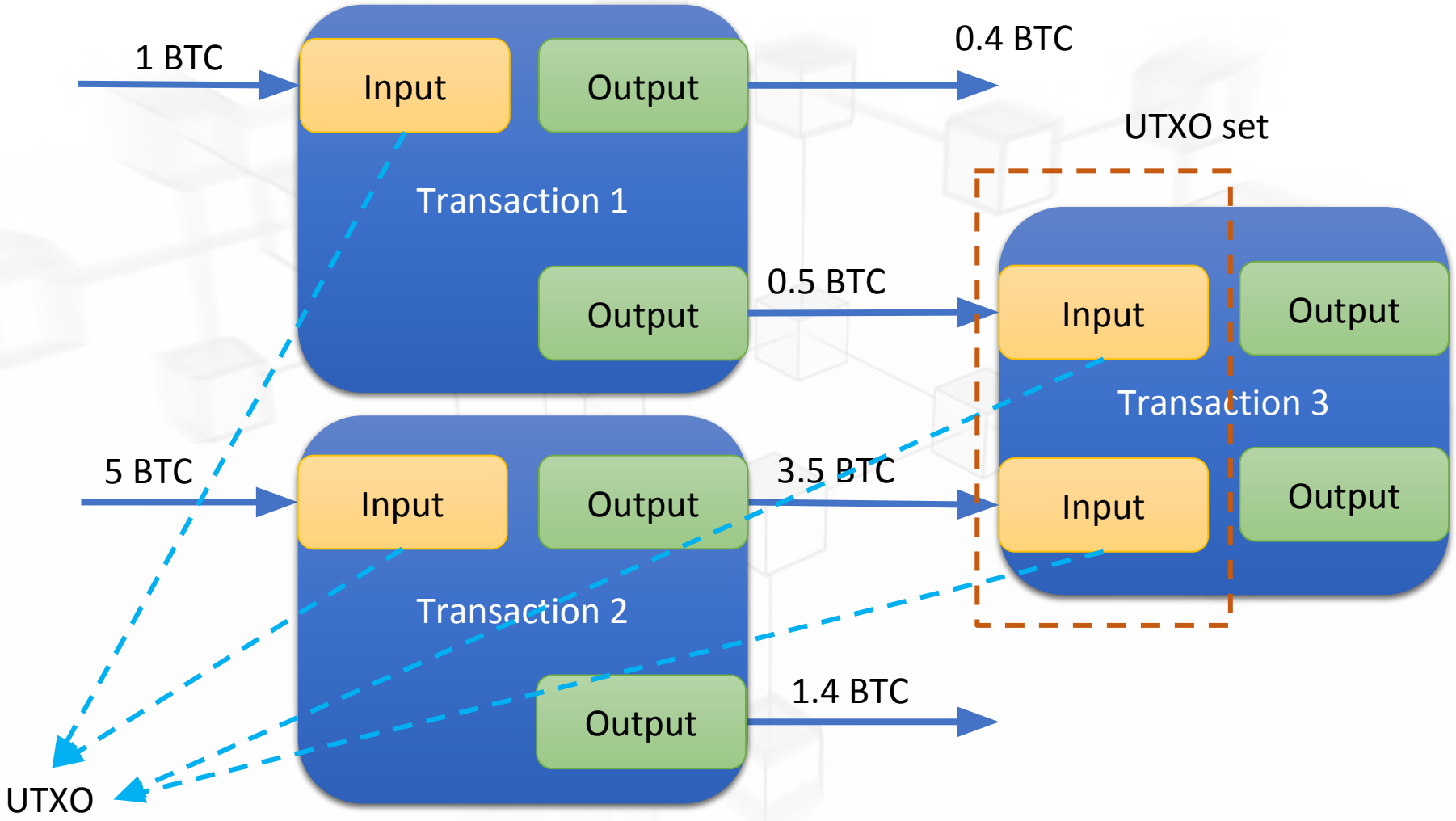
	Inputs	*0.55 BTC*
-	*Outputs*	*0.50 BTC*
	Difference	*0.05 BTC (implied transaction fee)*

Transaction Outputs and Inputs Transaction Outputs

- The fundamental building block of a bitcoin transaction is a *transaction output*
- Bitcoin full nodes track all available and spendable outputs, known as *unspent transaction outputs*, or *UTXO*
- The collection of all UTXO is known as the *UTXO set*
- The UTXO set grows as new UTXO is created and shrinks when UTXO is consumed.
- The wallet calculates the user's balance by scanning the blockchain and aggregating the value
- Most wallets maintain a database or use a database service to store a quick reference set of all the UTXO they can spend with the keys they control.
- Bitcoin can be divided down to eight decimal places as Satoshi's.
- Outputs are *discrete* and *indivisible* units of value, denominated in integer Satoshi's.
- An unspent output can only be consumed in its entirety by a transaction.

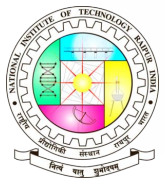


UTXO


1 BTC
Input
Output
0.4 BTC
Transaction 1
UTXO set
Output
0.5 BTC
Input
Output
Transaction 3
5 BTC
Input
Output
3.5 BTC
Input
Output
Transaction 2
Output
1.4 BTC
UTXO

Transaction Outputs

- Every bitcoin transaction creates output that recorded on ledger
- And create UXTO that is recognized by whole network and oner can use it for future transactions
- Transaction outputs consist of two parts:
  - An amount of bitcoin, denominated in *Satoshi's*,
  - A cryptographic puzzle that determines the conditions required to spend the output
- The cryptographic puzzle is also known as a *locking script*, a *witness script*, or a scriptPubKey.

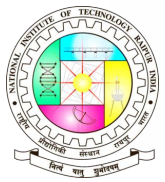


Transaction Outputs

```
"vout": [
{
  "value": 0.01500000,
"scriptPubKey": "OP_DUP OP_HASH160 ab68025513c3dbd2f7b92a94e0581f5d50f654e7 OP_EQUALVERIFY
     OP_CHECKSIG"
},
{
  "value": 0.08450000,
"scriptPubKey": "OP_DUP OP_HASH160 7f9b1a7fb68d60c536c2fd8aeaa53a8f3cc025a8 OP_EQUALVERIFY
OP_CHECKSIG",
}

]
```

- From above code, Each output is defined by a value and a cryptographic puzzle
- Bitcoin is recorded in integer in the transaction
- The second part of each output is the cryptographic puzzle that sets the conditions for spending.

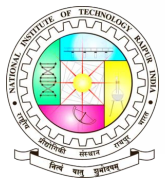


Transaction Outputs: Transaction serialization

- The transaction is exchange between two application in the serialized format
- That can be achieved by converting internal representation of data structure in to stream of bits
- This serialized format is used to encode data structure for storing in a file or transmission over the network

Table 6-1. Transaction output serialization

Size	Field	Description
8 bytes (little-endian)	Amount	Bitcoin value in Satoshi's (10-8 bitcoin)
1–9 bytes (VarInt)	Locking-Script Size	Locking-Script length in bytes, to follow
Variable	Locking-Script	A script defining the conditions needed to spend the output



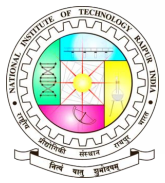
Transaction Outputs: Transaction serialization

0100000001186f9f998a5aa6f048e51dd8419a14d8a0f1a8a2836dd734d2804fe65fa35779000000
008b483045022100884d142d86652a3f47ba4746ec719bbfbd040a570b1deccbb6498c75c4ae24c
b02204b9f039ff08df09cbe9f6addac960298cad530a863ea8f53982c09db8f6e381301410484ecc0
d46f1918b30928fa0e4ed99f16a0fb4fde0735e7ade8416ab9fe423cc5412336376789d172787ec3
457eee41c04f4938de5cc17b4a10fa336a8d752adfffffffff0260e3160000000001976a914ab6802
5513c3dbd2f7b92a94e0581f5d50f654e788acd0ef8000000000001976a9147f9b1a7fb68d60c536
c2fd8aeaa53a8f3cc025a888ac 000000000

Serialized format of transaction shown above

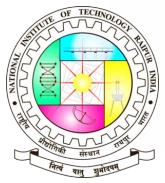
There are two outputs in the highlighted section, each serialized as shown in

- The value of 0.015 bitcoin is 1,500,000 Satoshi's. That's `16 e3 60` in hexadecimal
- In the serialized transaction, the value `16 e3 60` is encoded in little-endian (least-significant-byte-first) byte order, so it looks like `60 e3 16`.
- The `scriptPubKey` length is 25 bytes, which is `19` in hexadecimal.

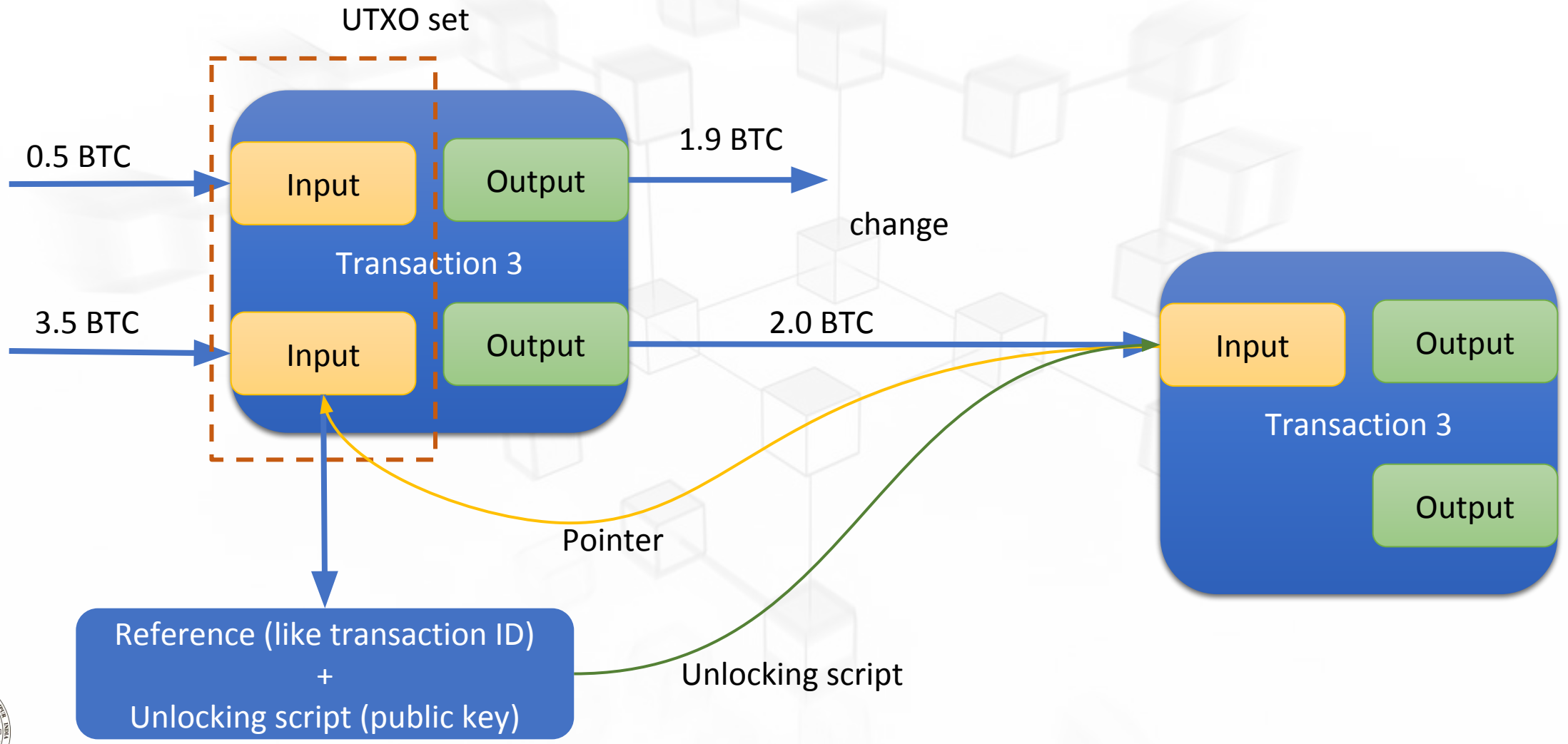


Transaction Inputs

- The transaction inputs identify by the reference number that UTXO will be consumed
- Also provide proof of ownership through unlocking script
- To make a transaction, wallet select UTXO having enough value to process a transaction request
- Sometime a single UTXO not having enough value so wall create a single input that pointing multiple UTXO and unlock with unlocking script
- There are two component of input
  - A pointer to an UTXO recorded in the blockchain.
  - An unlocking script, which satisfy the spending conditions
- Most often, the unlocking script is a digital signature and public key proving ownership of the bitcoin.



UTXO


UTXO set
0.5 BTC
3.5 BTC
Input
Output
Transaction 3
Input
Output
1.9 BTC
change
2.0 BTC
Input
Output
Transaction 3
Output
Pointer
Reference (like transaction ID)
+
Unlocking script (public key)
Unlocking script

Transaction Inputs

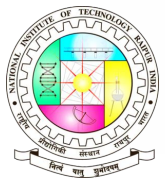
The transaction inputs are an array (list) called vin:

"vin": [
{

"txid":
"7957a35fe64f80d234d76d83a2a8f1a0d8149a41d81de548f0a65a8a999f6f18",
"vout": 0,
"scriptSig" :
"3045022100884d142d86652a3f47ba4746ec719bbfbd040a570b1deccbb6498c75c4
ae24cb02204b9f039ff08df09cbe9f6addac960298cad530a863ea8f53982c09db8f6e3
813[ALL]
0484ecc0d46f1918b30928fa0e4ed99f16a0fb4fde0735e7ade8416ab9fe423cc541233
6376789d172787ec3457eee41c04f4938de5cc17b4a10fa336a8d752adf",
"sequence": 4294967295
}
]

Transaction Inputs

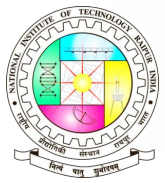
- The input contains four elements:
  - A transaction ID: referencing the transaction that contains the UTXO bein spent
  - An output index (vout): identifying which UTXO from that transaction is referenced (first one is zero)
  - A scriptSig: which satisfies the conditions placed on the UTXO, unlocking it for spending
  - A sequence number
- The sender need to retrieve UTXO used reference in the input
- All other validator nodes also need to retrieve the UTXO reference
- So transaction can be validated in further.



Transaction Inputs: Transaction serialization

Table 6-2. Transaction input serialization

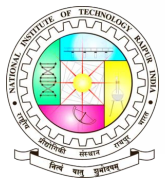
Size	Field	Description
32 bytes	Transaction Hash	Pointer to the transaction containing the UTXO to be spent
4 bytes	Output Index	The index number of the UTXO to be spent; first one is 0
1–9 bytes (VarInt)	Unlocking-Script Size	Unlocking-Script length in bytes, to follow
Variable	Unlocking-Script	A script that fulfills the conditions of the UTXO locking script
4 bytes	Sequence Number	Used for locktime or disabled (0xFFFFFFFF)



Transaction Inputs: Transaction serialization

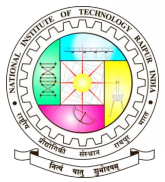
01000000001**186f9f998a5aa6f048e51dd8419a14d8a0f1a8a2836dd734d2804fe65fa35779**
000000008b483045022100884d142d86652a3f47ba4746ec719bbfbd040a570b1deccbb6
498c75c4ae24cb02204b9f039ff08df09cbe9f6addac960298cad530a863ea8f53982c09db
8f6e381301410484ecc0d46f1918b30928fa0e4ed99f16a0fb4fde0735e7ade8416ab9fe42
3cc5412336376789d172787ec3457eee41c04f4938de5cc17b4a10fa336a8d752adfffffffff
0260e31600000000001976a914ab68025513c3dbd2f7b92a94e0581f5d50f654e788acd0ef
800000000001976a9147f9b1a7fb68d60c536c2fd8aeaa53a8f3cc025a888ac00000000

- The transaction ID is serialized in reversed byte order, so it starts with (hex) 18 and ends with 79
- The output index is a 4-byte group of zeros, easy to identify
- The length of the scriptSig is 139 bytes, or 8b in hex
- The sequence number is set to FFFFFFFF, again easy to identify



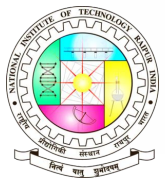
Transaction Fees

- May transaction implied a transaction fee that compensate the bitcoin miner
- Most Fees also serve as a security mechanism themselves to stop attackers to flood the network with transactions.
- Transaction fees are calculated based on the size of the transaction in kilobytes
- Overall, transaction fees are set based on market forces within the bitcoin network
- Any bitcoin service that creates transactions, including wallets, exchanges, retail applications, etc., *must* implement dynamic fees.
- Dynamic fees can be implemented through a third-party fee estimation service or with a built-in fee estimation algorithm

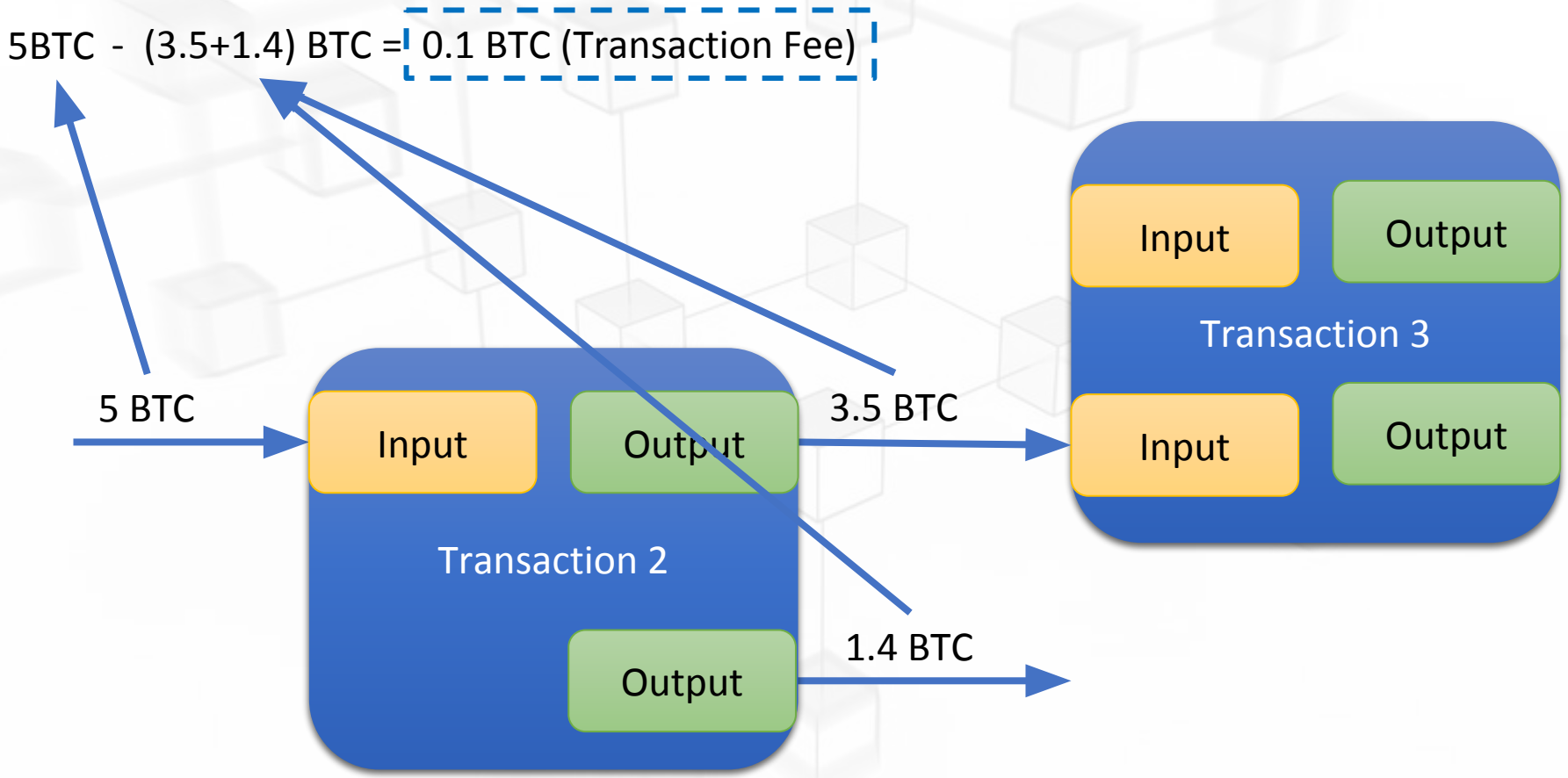


Transaction Fees

- Fee estimation algorithms calculate the appropriate fee,
- Based on capacity and the fees offered by “competing” transactions.
- fees are implied as the difference between the sum of inputs and the sum of outputs
- Any excess amount that remains after all outputs have been deducted from all inputs is the fee that is collected by the miners:
- Fees = Sum(Inputs) – Sum(Outputs)



UTXO


5BTC - (3.5+1.4) BTC = 0.1 BTC (Transaction Fee)
5 BTC
Input
Output
Transaction 2
3.5 BTC
1.4 BTC
Output
Input
Output
Transaction 3
Input
Output