**How does Counterparty work?**

Counterparty embeds data into regular Bitcoin transactions. To a regular Bitcoin client, these transactions look like normal Bitcoin transactions, with one party sending another party a very small amount of Bitcoin. A Counterparty node (which runs the Bitcoin client along with [the Counterparty client software](https://github.com/CounterpartyXCP/counterparty-lib)) will recognize and interpret the data in these Bitcoin transactions based on specific rules. From this, it constructs its own ledger of Counterparty transactions that it has seen on the Bitcoin network.

To better help understand this, [here](https://counterpartychain.io/transaction/c717d2d205155c2067786c08b7c7e6e6f904a18626969daca0ecb6c3e8bb7b8f) is a record of a Counterparty transaction where one address is sending 48 SJCX tokens (a custom token used by [Storj](https://storj.io/)) to another address. [Here](https://blockchain.info/tx/c717d2d205155c2067786c08b7c7e6e6f904a18626969daca0ecb6c3e8bb7b8f) is what this transaction looks like to a blockchain.info, a popular Bitcoin block explorer. You can see that while it is indeed a Bitcoin transaction, the amount of Bitcoin moved is small. In reality, the BTC spent is just enough to compensate the Bitcoin miners to include the transaction in a block. Essentially, the user that sent the transaction is paying the Bitcoin network to record and secure this embedded Counterparty data.

**So Counterparty is not its own Blockchain, but “rides on top of” Bitcoin?**

Yes. Another way to think of it is similar to a [Russian nesting doll](https://en.wikipedia.org/wiki/Matryoshka_doll), where the bigger doll would be the Bitcoin transaction, and the next doll (inside of it) would be a Counterparty transaction.

This embedding method is technically known as **embedded consensus**.

**Is Counterparty “polluting” the Bitcoin blockchain, then?**

No. 99%+ of Counterparty transactions utilize a data encoding method called OP\_RETURN, which is fully “prunable”, meaning that the data may be safely discarded by Bitcoin nodes which wish to do so. For the remaining 1% of transactions, an different encoding method is utilized that produces fully “spendable” outputs. These outputs do not stick around in the critical list of unspent outputs (the “UTXO set”).

On top of this, every Counterparty transaction pays a fair fee to the network for inclusion.

**Are Counterparty transactions less secure than Bitcoin transactions?**

As Counterparty transactions *are* Bitcoin transactions, their data is proably just as secure as any other Bitcoin transaction.

**How do the Counterparty nodes stay in sync? What’s to stop one node from disagreeing with another?**

As all Counterparty nodes run the same code, and all receive the same Bitcoin transaction data, the ledgers across each node match exactly. Counterparty nodes are not like Bitcoin nodes in that they don’t communicate with each other: they simply connect to the Bitcoin software and download transactions from it, decoding each one as they go along. In this way, the immense security and computing power behind Bitcoin is leveraged as the “transport network” for Counterparty data.

Given the above, there is no “Counterparty peer to peer network” like there is a “Bitcoin peer-to-peer network”: Counterparty-aware nodes comprise a subset of the Bitcoin full nodes in existance.

**What about Sidechains?**

Counterparty is optimal for mainly higher value transactions and greatly benefits from the security of the main chain. However, if sidechains are ever released, there is no reason that they couldn’t be made to work with Counterparty. This is the beauty of Counterparty’s embedded consensus technology – it can work with just about any blockchain out there, including sidechain designs.

**What kind of addresses does Counterparty use?**

*Exactly* the same Bitcoin addresses we all know and love. As such, Counterparty tokens (such as XCP, SJCX, CAKE, and more) may be sent to *any* Bitcoin address.

**What is XCP?**

XCP is the native token of Counterparty. It is a technical necessity for adding advanced features to Counterparty, which by nature require a protocol aware currency. Bitcoin can only be aware of BTC, while Counterparty can be aware of both BTC and XCP itself. This makes it possible to escrow funds, trade in a decentralized manner, and harness the full potential of programmable money.

*To learn more about XCP, see*[*about XCP*](https://counterparty.io/docs/FAQ-XCP)*.*

**Can I secure my XCP and tokens in cold storage?**

Yes. You can make a regular Bitcoin paper wallet and store them there. Later, you can sweep the funds into a Counterparty wallet, like Counterwallet.

Counterwallet also supports the use of Offline Armory. More info on that is [here](https://counterparty.io/docs/create_armory_address).

**Is a 51% attack against Counterparty possible?**

As every Counterparty transaction is a Bitcoin transaction, to do a “51% attack” on Counterparty you would have to do a 51% attack on Bitcoin.

**Besides a 51% attack, what are the other risks to consensus?**

The Counterparty network could be effectively “forked” by a sizable number of people running different versions of the Counterparty client that had different “consensus sensitive code” (i.e. protocol code). In this case, if a transaction was read in from the Bitcoin client software, the differing code may cause two different interpretations of the data, and thus, two different ledger states.

As long as all participants run software that has the same protocol rules (even if it is different Counterparty client implementations), this situation will not happen. The reference client includes extensive safeguards that help detect and prevent this from happening.

That being said, [the Counterparty client](https://github.com/CounterpartyXCP/counterparty-lib) is completely open-source. Anyone is able to copy the code and make their own modifications. They can then run their modified version of the software, which technically may generate a different ledger than everyone else. This is similar to Bitcoin itself. However, to have any impact, that person would have to get others to run it, who would have to trust this individual more than they trust the Counterparty development team. This new ledger would not be “Counterparty”. It would be a separate ledger with its own protocol rules. Services built on this ledger (such as a block explorer) would not agree with similar services built on the Counterparty ledger.

**So can the Counterparty Team rewrite the Counterparty ledger’s history, in an emergency or by decree? How does that compare to the same risks with Bitcoin Core devs?**

It’s identical to the case with Bitcoin. The Bitcoin core devs could publish a copy of Bitcoin Core that does anything, but no one would download it.

Counterparty is 100% open source, with [a list of code changes](https://github.com/CounterpartyXCP/counterparty-lib/releases) from one release to the next visible for all to see and inspect.

**What about support for other blockchains instead of Bitcoin?**

Counterparty is built on Bitcoin. That has always been the case and we do not see it changing, ever. For other blockchains, there are “forks” of the Counterparty software. Examples would be Dogeparty for Dogecoin, and Viacoin’s ClearingHouse. We generally encourage forks on other blockchains, especially if they help contribute back bug fixes and enhancements to the main Counterparty codebase.

**What is Bitcoin fails or becomes co-opted?**

In the event of a catastrophic failure of the Bitcoin network, Counterparty *does* have the technical capability of “freezing” balances and migrating to another blockchain, like Litecoin for instance, with relative ease.

**What happens if and when OP\_RETURN data is auto-pruned?**

Counterparty only needs some Bitcoin full nodes somewhere to have an unpruned copy of the blockchain. As every Counterparty full node is also a Bitcoin full node, this is easily done.

**How are blockchain reorganizations (“reorgs”) handled by Counterparty?**

Blockchain reorganizations are essentially handled by Counterparty the same way they are handled by Bitcoin. If the Counterparty software detects that a reorganization has occurred, it will utilize an internal “undolog” to quickly undo (roll back) transactions up to the point of the chain branching, and then process new transactions on the now-longest chain.

**How can a thin client trustlessly lookup the Bitcoin public address associated with the OSTOCK asset name?**

You can use a local copy of the blockchain just fine. The only difference between Counterparty and Bitcoin here is that Counterparty doesn’t support SPV. We’re working on solutions to this issue. Protocols like VerSum offer excellent models for untrusted verification here.