Surviving the Upcoming Bitcoin Fork Split

How to survive the Fork Split, a Guide for Exchanges and Businesses

Jerry Chan
jerry.d.chan@bittoku.co.jp
March 6th, 2017

12A8LKsf2qHnF95JXBmui8yBteestqquGz

Abstract

With the Bitcoin community potentially at a cusp with some proponents favouring supporting their own scaling solution at all costs at the expense of a system wide consensus or compromise, the possibility of a fork split in Bitcoin is at its highest levels since inception. One thing is for certain if the current deadlock between different proposed scaling strategies persists, then the only thing that would result is the inability for Bitcoin to scale, and in that light, a fork split although unfortunate, would be the best outcome for all parties involved, that so much is for certain. In this light, this document will assume that a fork split is thus likely to occur, and attempt to address the technical and economic ramifications of that split on businesses and users operating on the Bitcoin blockchain.

The justification for this document is due to the recent suggestions from the Core development community about an implementation of a user activated soft fork, (one that can be executed with a minority of hashpower supporting it) this presents a clear and imminent danger to the stability of the network especially for the exchanges and businesses that run on Bitcoin. Due to the potential of the chain splitting into 2 or even 3 forks in the case of a User Activated Soft Fork (UASF), businesses and especially exchanges will be in danger of losing customer funds if they are not prepared to handle the situation especially if a UASF were to happen without warning. The purpose of this document is to prepare such businesses so that they can continue to protect their customers funds in such an event, regardless of whether they intend to actually support multiple fork coins or not.

Types of Fork splits

First we should enumerate the types of splits that can happen.

Soft Fork

A fork in which a new consensus rule is added, which is a subset of the previous rules. Blocks conforming to the newer more strict rules are valid in the new chain, and the old chain, while blocks produced by the old chain under the previous less strict rules are not valid in the new chain.

Hard Fork

A fork in which a consensus rule is removed or relaxed. Blocks conforming to the new consensus rules are valid in the new chain only, and not on the old chain, while blocks produced by the old chain are valid in the new one.

Minority Fork (Hard/Soft)

This is when a minority hashrate supported chain deliberately forks the network by publishing a block which the majority will not accept. Conversely, the minority chain can arbitrarily create a rule, enforced only by them that makes all the future blocks produced by the majority chain invalid (User Activated Soft Fork).

Majority Fork (Hard/Soft)

This is when a majority hashrate supported chain deliberately forks the network by publishing a block that the minority hashrate supported chain will not accept. Alternatively they may decide to activate and enforce a new rule which the minority chain will not enforce and be ambivalent to.

Forced User Activated Soft Fork (UASF)

This type of soft fork is done when a majority of the user nodes (wallets, home node) upgrade to a version of the software which coordinates a block height after which a certain new validity rule is enforced, for example a rule that would say that a block which an UTXOs that isn't a paid to a new SegWit output is invalid. This can be done ***without*** a majority of hashpower supporting it.

This type of fork has the potential for customers to not see withdrawals from exchanges and create customer support issues. Exchanges intending to support both forks should upgrade their node for this chain and ensure that withdrawal requests are processed through this node for the clients who demand it. Any support of this policy implicitly means that the exchange is agreeing to support split coins products in the future. This means an remaining balance on the other chain for the client must be created and maintained within the exchanges accounting system. It would be advised that the exchange split the coins before processing any withdrawals to avoid confusing balances. (see splitting process below)

Technical features of forks

Reorg risks

The reorg risk is the risk that sometime after the fork one chain which was formerly the minority chain may somehow overtake the majority chain, and thus cause a reversion of that chain back to the minority chain, undoing all the transactions that were in it. This actually happens naturally as part of normal Bitcoin network operations and reorgs of a couple blocks a few times a week is not uncommon. The difference in the case of a split fork is that the reorg risk is one-sided.

Soft forks create a reorg risk on one fork side only (the other chain) because of the nature of a soft fork, namely that it is an added restriction in the validity rules for a block, it will never accept a longer chain produced by the other fork as it will deem it invalid. The converse is not true, because a block that adheres to the additional validity rule will still be valid on the original chain. Therefore there is a increasing chance (so long as the hashpower is greater) that the new more restrictive chain may out pace the original chain and cause a reorg on it. The converse is true for a Hard fork, which has a reorg risk on its *own* chain but not on the other chain. That is because blocks which are created by the other chain will always be valid on the hard forked

one, but the reverse is not true, so a hard fork creates a chain which can be potentially reorg'd without the reciprocal risk on the other chain.

This risk is only a factor if the more restrictive chain has more PoW hashing power by a large enough margin such that it can outpace the original chain's hashing power, and thus find more successive blocks in a row than the original chain. This reorg risk is something that the original chain must accept. If this risk is deemed too excessive it may be possible that a one-time checkpoint be created after the Forking Block so that it cannot be reorg'd beyond this checkpoint. One possible way of implementing this would be a soft-fork rule added which would make any block which does not contain the Forking Block as an ancestor would be deemed invalid.

Split Coin Assets

Coins in existence at the point of the fork will become 1 of 3 possible types, pre-fork, post-fork chain_A, or post-fork chain_B. It is easier to think of this in terms of UTXOs. A pre-fork UTXO can become a post-fork A or B UTXO, but not vice-versa (a post-fork A, or post-fork B UTXO cannot be reverted to a pre-fork UTXO). These resulting UTXOs can only exist on one chain or the other but not both.

A pre-fork UTXO may continue to persist, and be continually passed around post fork, on both fork A and B, so long as they are not mixed via being used in a transaction that include a post-fork UTXO as another input. Once mixed in such a way and confirmed in a transaction on either Fork A or B, then the UTXO is said to be permanently split.

Concerns of additional "value" created

This is the issue which made some people very concerned when Ethereum underwent a split. They were very well elaborated on in this Coindesk article. In summary though it is not possible to execute these quick 'arbitrage' opportunities in the case of a Bitcoin split, because splitting coins is not easy and require quite a bit of work on the part of any exchange or parties acting as one. Additionally, the supply of pure *Chain A* coins and *Chain B* coins will be very low initially and accumulate at a very slow rate. (the rate is governed by the natural diffusion rate of newly mined coins in addition to the efforts of direct *coin separation* done by exchanges via the process described below) This prevents the massive speculative bets that were made on ETC which ended up losing a lot of money for the risky speculators who thought that if they all bought into ETC then most of the miners would follow (and thus legitimize their holdings and wagers).

Double Spend Attacks

This is often quoted as a problem with a blockchain split, namely that any transaction valid on one chain is valid on both and thus there is an increased chance of double spending in the case of a split. This is achieved by taking advantage of the fact that your counterparty may be on one chain while you are on another. That way you can spend on one chain, but then double spend

the same UTXO on the other chain to pay yourself. This is a false attack, as the person who you are paying is generally not going to be interested in receiving payment on both chains anyway.

Transaction Replay

This sort of vulnerability is a variation of the double spend attack, and was first explained prior to the Ethereum split into ETC and ETH, and it affects mostly exchanges who were ill-prepared to handle both forks. It was explained in an article here, and it is possible to protect against this sort of attack with proper preparation. In summary, exchanges wishing to support both chains post split and to facilitate trade between them as separate products needs to keep the balances of each fork separate. In addition coin-separation processes (described below) must be put in place so that exchanges do not inadvertently send out any coins unintentionally. This is especially important to those who wish to employ a coin separation strategy which manages the separation at the *time of withdrawal.*

Considerations - Bitcoin Currency Exchanges

Supporting a split

The general strategy to support both coins from an exchange is divided into 2 parts, 1) **procurement of split coins**, and 2) **management of a coin separation process**. In addition to this all businesses should run both node clients of each fork in order to monitor activity in the both chains, in order to prevent loss of customer funds.

Procurement of Split Coins

Split coins are needed in order to separate customer deposits and to ensure withdrawals are not mixed. The exchange can get split coins either by getting pure coinbase generated coins directly from miners, or splitting coins themselves.

Procurement of Pure Coinbase

Buy coinbase straight from the miners from both forks chains, keep these in separate ‘pure’ coin pools (addresses), different for each fork. These source pools of pure post-fork coinbase coins will be essential in the process of coin separation. You must verify the provenance (history) of each UTXO to ensure that it is derived from a coinbase txn from the respective chain ***after the forking block.*** Be aware that the exchange only needs to procure one sample of a pure coinbase UTXO, as the exchange can then use this initial sample to convert its entire inventory to separated chain_A and chain_B coins. If an exchange does not want to wash its entire inventory through a splitting transactions, or cannot afford to, then it is advised that the exchange NOT support both chain coins separately.

Creating Split Coins Manually

The method to manually split coins is described here, and can be employed to generate split coins without the need of miner coinbase coins. Once generated, the needed supply of split coins can be maintained by creating new split coins with the exchange's own inventory whenever necessary.

Coin Separation Management

Exchanges intent on supporting both forks should split their hot wallets into 3 buckets, *chain_A*, *chain_B*, and *pre-fork* pools. Management of both coins as separate products on the trading platform centres around the separation of the coins at point of deposit into the exchange, and ensuring that withdrawals are processed using only separated coins from the respective pools.

Method 1 - Split on Deposit

For every deposit from customers, send some coinbase into that address from each pre-fork coin pool, of the some small amount, after confirming (your preferred number of confirmations that you wait) create a txn in both nodes moving all the balance (as to ensure that you include the coinbase tinting UTXO as well) into a separate address for the customer in that specific chain. The coins have now been separated. *the same address can be used for both coins but that may lead to confusing for accounting software as you will be relying on block explorers to know which chain the balance for any given address came from in order to determine which fork balance to represent.

Deposit Policy

2 different deposit addresses should be given to customers for deposits if the support of both forks is desired. Although these addresses are different, you should prepare for the case where the customer deposits coins into the wrong address and you need to split them, or if they deposit into both. Note that this may result in a client depositing coins and actually be credited with *BOTH* chain_A coins and chain_B coins at the same time. This will eliminate the issue with clients who accidentally deposit prefork coins and then later demanding withdrawal of both coins.

Withdraw Policy

Exchanges deciding to support both forks must be aware that withdrawals must be processed on each chain separately. This means that even though the user may have deposited pre-fork coins, the exchanges must separate them into chain_A and chain_B coins internally, and treat withdraws only with respect to chain_A or chain_B coins.

Method 2 - Split on Withdrawal

An alternative method of coin separation management would be to handle deposits the same way, but instead of splitting the coins on deposits, simply determine which coins were

deposited, and credit the balance of the customer to that effect. Keep the deposited coins (split or prefork) together in the same addresses, and only upon withdrawal are the coins separated.

Deposit Policy

Continue to use 1 address for customer deposits. This may be a multisig address or just a regular address depending on the exchanges security model. On deposit, monitor whether or not the deposit shows up on chain_A, chain_B, or both. If they credit (or refund) the customer accounts as appropriate. Note that this may result in a client depositing coins and actually be credited with *BOTH* chain_A coins and chain_B coins at the same time. This will eliminate the issue with clients who accidentally deposit prefork coins and then later demanding withdrawal of both coins.

Withdrawal Policy

On withdrawal, send some nominal amount of split coins (of the appropriate chain) into the customer's deposit address to ensure that the withdrawal transaction cannot be replayed on the other chain. This effectively separates the coins on withdrawal.

Exchange Customer Considerations

Existing coin balances policy

What happens to BTC balances (which are off chain) after a fork and an exchange decides to support both forks? Customers should be warned that should a fork occur, that the exchange will preserve the value of the coins in any forks created, but that in order to do so withdrawals will be withheld until the forks resolve back into 1 chain, or 2 chains emerge and persist past the point of reasonable reorg risk. (Each exchange would have to determine at which point they feel comfortable with assuming this risk, but a good estimate would be 100 blocks deep on the minority chain)

Any remaining coin balances at that point should be separated by mixing with coinbase coins from each chain and separated into 2 different accounts for each client, one for each chain fork coin. 1 pre-fork coin should equal 1 post-fork coin for each post fork chains supported. There should be no more pre-fork coin balances kept at this point forward

Wallet considerations

if dual coins are being supported, and withdrawals on one coin are supported then it must be stressed that the users receiving split coins must be using wallets which are running on a node which will recognize the split.

SPV wallets will see potentially conflicting confirmations (at different block heights) for the same transaction depending on which nodes they are connected to. This shouldn't matter as far as confirming the coins are received.
Wallets which are running on one chain or the other will only see the transaction confirmed if it is watching the correct chain.
All wallets will be able to see the transaction (0-conf) if it is a prefork UTXO being spent. If it is a split coin, then it will not see the transaction if watching on the other fork.

Customer support requests for withdrawals on one chain not showing up on their wallet should be expected. This will likely be a non issue, for retrieval of the coins will be as simple as changing the wallet to read blocks from the other chain. No coins are actually lost, they are just not visible.

Not Supporting a Split

Support longest chain only. This is the simplest option as no specific process to separate coins need to be supported. Process withdrawals and deposits only one fork, ignore the other. *However* this puts customers funds *at risk*.

Even if an exchange has no intention of publicly supporting the trading of both fork coins, if both chain split forks persist past 100 blocks, exchanges still have the duty to process withdrawals for clients that expect to own both chain coins. Supporting this policy means to follow the **coin separation management** policy as described above in order to make available the forked coins to be withdrawn from the exchange.

This policy is safe, as long as it is made aware to the customers. Even though an exchange has no intention of supporting both fork coins, there is the potential issue that customers who did not know this policy and tried to deposit chain_A coin into the exchange, which the exchange does not recognize or credit. How this will be treated is up to the policy of the exchange, but in the interests of protecting the funds of the customers, each exchange should adopt a coin separation policy as described above under the "Supporting a Split" section.

Considerations - Wallet Users

Users have the option of totally ignoring the split or trying to capitalize on the split and sell the split chain coins that they do not support for money. Keeping your coins separate would require using a wallet that would support the chain. Whether or not a wallet supports the minority fork will depend on the wallet.

Web Wallets

You are at the mercy of which chain the web service will support. It will be doubtful that they will support both, most likely just the longest chain. They may be able to provide the ability to choose which chain you want to connect to should they decide to support both as a service.

SPV Wallets

SPV wallets by default will not be aware of the block size and thus just get the highest block height. Unfortunately if the SPV wallet connects to ALL nodes from one chain only, then it will report a balance that may differ from if it were to connect to all nodes from other chain. If it connects to a mix of nodes, then it will report the balance on the longest chain.

Hardware wallets

This depends on the specific wallet as some can be paired with your own node while some use a server backend. Find out which category your hardware wallet belongs to and refer to that wallets section for further advice.

Server based wallets

If you can control which node the wallet points to, you will have to direct your wallet to use a known node that is on the minority chain. If you cannot control the server that your wallet connects to then you will be locked into transacting on the chain that your wallet provider is supporting. They may be able to provide details in order to choose which chain you want to connect to should they decide to support both.

Node client wallets

The easiest of the options, you can just run the software of the chain you wish to connect to.

Considerations - Businesses and Payment Processors

As a business, the best option is to follow the longest chain. Supporting both chains will be confusing to your customers and cause support issues as customers accidentally send you prefork coins and you would be required to refund them back separated coins on the fork that they did not intend to pay with. Unlike exchanges, supporting the minority chain does not profit your business as it does an exchange which can stand to make more trading fees. Therefore supporting the minority chain is only a net cost to your business. But if you want to support both chains for ideological reasons, follow the same instructions on keeping your coins separate in the Exchange section. One recommendation is to run a client node for each of the major competing forks in order to maintain visibility into the other chain. This will assist in managing customer satisfaction in the case where clients accidentally pay with the wrong coin, and a refund needs to be processed. This is a necessity for merchant payment processing businesses.

General Fork Risks

UASF Risk

The employment of UASF in order to activate a fork without the majority of the hashpower carries with it some inherent risks that a normal majority fork does not. Namely, that it opens up the possibility of creating up to 3 forks at least temporarily.

One possible scenario is that if the UASF is initiated by an influential party resulting in a minority soft fork activating SegWit, then this may trigger a third hard fork being created from the remaining network due to the latent demand for a hard fork which would increase the block size limit (for example, to 4MB). This is made possible by the initial UASF because if presumably all the supporters of SegWit would have forked off into their own chain, the proponents of a block size increase hard fork would no longer have any opposition and the hashpower remaining which would support the hard fork would be in the majority, sufficient to trigger a safe hard fork. This would result in 3 forks, a SegWit fork, a 4MB fork, and the original chain. This 4MB fork would likely create a checkpoint post forking block to eliminate the reorg risk from the SegWit fork, leaving only the original chain vulnerable, and thus putting an additional incentive for remaining participants to join either the 4MB fork or the SegWit one. Nevertheless, for a short time there may be 3 chains live at a given time, which makes it even more important that businesses prepare their systems for this possibility in order to protect the value of their customers deposits.

Speculation Risk

Whenever a chain splits resulting in a (potentially temporary) new coin balances, there will be a some traders who will try to capitalize by buying up the supply of the minority (cheaper) split coin in the attempt to profit from the exchange, or selling all the supply of one split coin and buying the other (if one has a view on which way the fork will resolve). This is a *HIGHLY RISKY* activity and should be avoided unless one is prepared to lose all their value. The safest way to preserve your value during a coin split is to not move any of their coins until the split has been resolved, either by collapsing back into one chain, or with 2 stable new chains emerging.

Glossary

Forking Block

This is the block that causes the fork split in the chain. Presumed to be the first block which is viewed as invalid to one side of the network, but valid to the other.

UTXOs

These are unspent transaction outputs, generated by transactions. The set of UTXOs in a nodes memory is the sum total of all the coins available to be spent in the system (excluding unspent coinbase generated coins)

Prefork UTXOs

These UTXOs were created from blocks leading up to, but not including, the Forking Block. These are sometimes referred to as ‘mixed UTXOs’ because they can be used on both chain_A or chain_B.

Chain_A, Chain_B, collectively ‘post-fork UXTOs’ or ‘split coins’

These UTXOs were minted in coinbases from blocks starting from the Forking Block and afterwards. If an UTXO can trace back to a coinbase generate transaction that is minted from the Forking Block or afterwards, they are considered ‘split coins’. As there are 2 sets of them, one for Chain A and one for Chain B, they should be treated separately and for all intents and purposes, they are separate coins.

Coins

For the purposes of this document, the term is sometimes used to be synonymous to UTXO, as the context will indicate.

Coinbase

Otherwise knows as “Generation transactions”, these are coins which are created directly from the minting of the block as their prior transaction

Node

Client software which connects to and participates on a given consensus network. This is sometimes used synonymously to mean client node software with a block explorer.

Previous Publications

Hard Fork Risk Analysis
http://www.wallstreettechnologist.com/2016/11/18/hard-fork-risk-analysis-if-the-worse-happens-how-bad-can-it-be/

Emergent Consensus - a Guide to Forking Safely
http://www.wallstreettechnologist.com/2016/10/14/emergent-consensus-guide-to-forking-safely/

Lightning Network - Will it Save or Break Bitcoin?
http://www.wallstreettechnologist.com/2016/10/03/lightning-network-will-it-save-bitcoin-or-break-it/
Bitcoin XT and the Hard fork that will split us all
http://www.wallstreettechnologist.com/2015/08/19/bitcoin-xt-vs-core-blocksize-limit-the-schism-that-divides-us-all/