A possible side chain without Bitcoin Software modification

By Phil Champagne

BookOfSatoshi@gmail.com

July 22nd, 2014

At this point, to insure both security and stability, the Bitcoin community is reluctant to conduct any experiment, large or small, or introduce new features to Bitcoin. The side chain concept would incorporate the capability of running a separate block chain in Bitcoin, thus providing the ability to perform experiments and allowing for an entirely new set of services and benefits. So far, whenever important changes are considered, a new altcoin has been created, each with its own block chain, protocol, and currency. Currently Bitcoin has an extensive network of miners hosting a significant, combined hash power from which no altcoin can benefit in any manner. Ideally, Bitcoin's existing currency and CPU hash power could be used were side chains (i.e., additional block chains) to be allowed. However, proposals so far require a modification of the existing Bitcoin software to allow for the existence of side chains.

This proposal describes a method of supporting side chains as defined below that would require no modification to the current Bitcoin protocol:

1. A side chain allowing for public storage of documents whose integrity would be maintained but no handling of currency as part of transactions recorded on it.
2. A side chain with transactions using bitcoins which have been transferred via one-way peg.
3. A side chain with transactions involving its own currency but still using the hash power of the Bitcoin network.

In all the cases listed above, operators (e.g., miners) of the side chain would be remunerated in BTC handled on the Bitcoin's main block chain.

For the features and services supported on this side chain to have the highest versatility, the side chain would have to be capable of handling currency transactions it would record itself. But in order for this currency to be in bitcoins rather than its own distinct currency, a transfer of bitcoins from the main block chain to this side chain would have to occur. In other words, for any given bitcoins (currency) handled on the side chain, a corresponding amount would have to be "frozen" or "suspended" on the Bitcoin block chain. In effect, bitcoins transferred for handling on the side chain could not also be available for transactions on the Bitcoin block chain as well; otherwise we end up with distinct currencies. So far, two methods have been proposed to achieve this, one involving a one-way peg, in which bitcoins move to the side chain forever, and another involving two-way pegs in which bitcoins can move back to the main block chain at some point. One-way peg is not ideal as it forever restricts the transferred bitcoins to the confined area of the side chain. Further, should an experiment involving the side chain fail, those bitcoins re-allocated to the side chain would become forever worthless or lost. On the other hand, two-way pegging would provide the ultimate method to achieve the aims of having a side chain but would require a modification to the Bitcoin protocol. Such a modification would require a relatively significant level of support and so would take time to adopt. Talks are currently ongoing on ways to introduce those changes which are still being worked on. In fairness, we should mention that a two-way peg could be performed without modifying the current Bitcoin protocol by having a centralized third party (or a semi-centralized escrow service using multisig) "freezing" the bitcoins on the main block chain while they are in use on the side chain. The major drawback of this approach is that it requires trust in this third party. The proposal described herein would allow for the immediate inclusion of a side chain without the existing Bitcoin software needing to be modified.

The features and services associated with a side chain might not need to involve currency transaction. Rather, the side chain could be used as a public storage mechanism for documents such as mortgages or liens. The drawback of this proposal is that, were bitcoins to be handled on the side chain, only one-way pegging could be supported. In this case, the side chain protocol could mandate that operators of the side chain be compensated out of those transferred bitcoins in the same manner as currently on Bitcoin. However, it is very likely that the type of work required of operators to support these new features would require substantially more bandwidth and/or disk storage. Hence a sufficient number of bitcoins would need to be transferred to the side chain to properly incentivize operators (i.e., miners) to perform their work.

One-way pegging could be achieved by defining a specific Bitcoin address as a location to which bitcoins on the main block chain go to "die". It is a Bitcoin address for which no one has the corresponding private key such as "111111bitcoinblackholeforsidechainABC". A user desiring to transfer bitcoins to the side chain would sign a transfer transaction record using the private key corresponding to the Bitcoin address used to transfer those bitcoins.

Another issue with compensating side chain operators with one-peg bitcoins is the "altered" value those bitcoins would have as a result of being tied to the new side chain. Therefore, we propose an alternative in which operators of the side chain ("miners") are compensated in BTC native to the primary Bitcoin block chain. The side chain would therefore be dependent on the Bitcoin network and main block chain while the Bitcoin network would be unaware of the side chain's existence, thus reducing the number of bitcoins that might need to be transferred to the side chain for those side chain transactions involving Bitcoin currency. An alternative proposal could incorporate a side chain having its own currency while still leveraging Bitcoin's substantial network of miners and hash power. In such case, operators could be rewarded with the side chain currency or with bitcoins as proposed earlier.

Compensating operators

Bitcoin's block chain employs the proof-of-work to secure its integrity over this decentralized distributed peer-to-peer network. Just as with Bitcoin's block chain, a side chain would also need to have its data secured and its integrity maintained. Bitcoin uses its own currency as a method to reward those nodes winning the proof-of-work. As opposed to altcoins, which are operating under entirely new protocols with their own currencies, the side chain we propose will still use BTC on the Bitcoin primary block chain to reward nodes for processing and handling transactions on the side chain which involve bandwidth usage and disk storage. How then can we maintain the side chain's integrity using BTC on the block chain without modifying the existing Bitcoin protocol? At its core, what we need is a way to compensate the nodes in BTC native to the primary Bitcoin block chain for maintenance performed on the separate, side chain network.

Just as with Bitcoin, a single node would be responsible for creating the next block of the side chain. With Bitcoin, the node ("miner") creating a block is compensated out of the same block by "minting" new bitcoins as well as being awarded the aggregate transaction fees of the transactions comprising that block. Those fees are not sent to a particular Bitcoin address; rather, the winning node ("miner") simply adds the sum of those fees to a Bitcoin address of his choosing. Bitcoin's block chain acts as the system's ledger of accounts and, since that node is responsible for updating it, he can easily make the accounting modification in favor of his own Bitcoin address as part of creating the new block. In our current proposal, we cannot assume that the node creating the latest block of Bitcoin's block chain would also be involved with the side chain unless the Bitcoin protocol were updated to allow it. Therefore, a method is needed whereby the node creating the latest block of the side chain would be compensated in bitcoins even though this node could have no involvement with the Bitcoin's protocol, other than being a passive user receiving and sending Bitcoin transactions.

Users of the side chain would send a transaction containing a transaction fee. Note that this side chain might not necessarily be handling currency transactions but rather general information such as a contract to be recorded publicly. So how precisely can the operator ("miner") that has been selected to create the next side chain block be compensated out of the transaction fees sent on the Bitcoin network? Compensating the side chain operator would require sending bitcoins using regular transactions on the Bitcoin network, and this could be accomplished only by sending them to this node's Bitcoin address.

Thus, two networks would be involved. The Bitcoin network would be used by the side chain's users as a way to compensate the side chain nodes. The second network would be the side chain, on which the side chain transactions would be sent. The side chain nodes would only handle side chain transactions for which transaction fees have been sent on the Bitcoin network. Given that up to about ten minutes are needed for a transaction to be recorded in the Bitcoin network and at least six blocks (or 60 minutes) before firm confirmation is received, how then can a side chain's node be compensated immediately as is the case in the Bitcoin network?

The side chain operators would not expect immediate payment; rather, they would prefer the certainty of compensation. Also, it would not matter to these operators who is actually compensating them. Therefore, one solution would be to have them ask to be compensated **after** they create the block. The winning node would be adding a Bitcoin address that it owns to the side chain block being created, thus indicating to users where to send their payments. One approach could have the users of the side chain paying **after** their block has been recorded. In other words, users would send their transactions, wait to see when they are recorded in the latest side chain block, look up the Bitcoin address required to send payment to, and make payment on the Bitcoin network for the required transaction fee. A possible complication arises in the case of non-payment; were this to occur, would the side chain network need to remove any transactions for which no corresponding payments had been made?

However, there exists a still easier approach. Is it that crucial that the operator who included a given transaction in the side chain block be the very same one being compensated by the owner of that transaction? I do not believe so, and relaxing this requirement would allow users whose transactions are to be recorded in the next block compensate the winning node that created the prior block (see Figure 1).

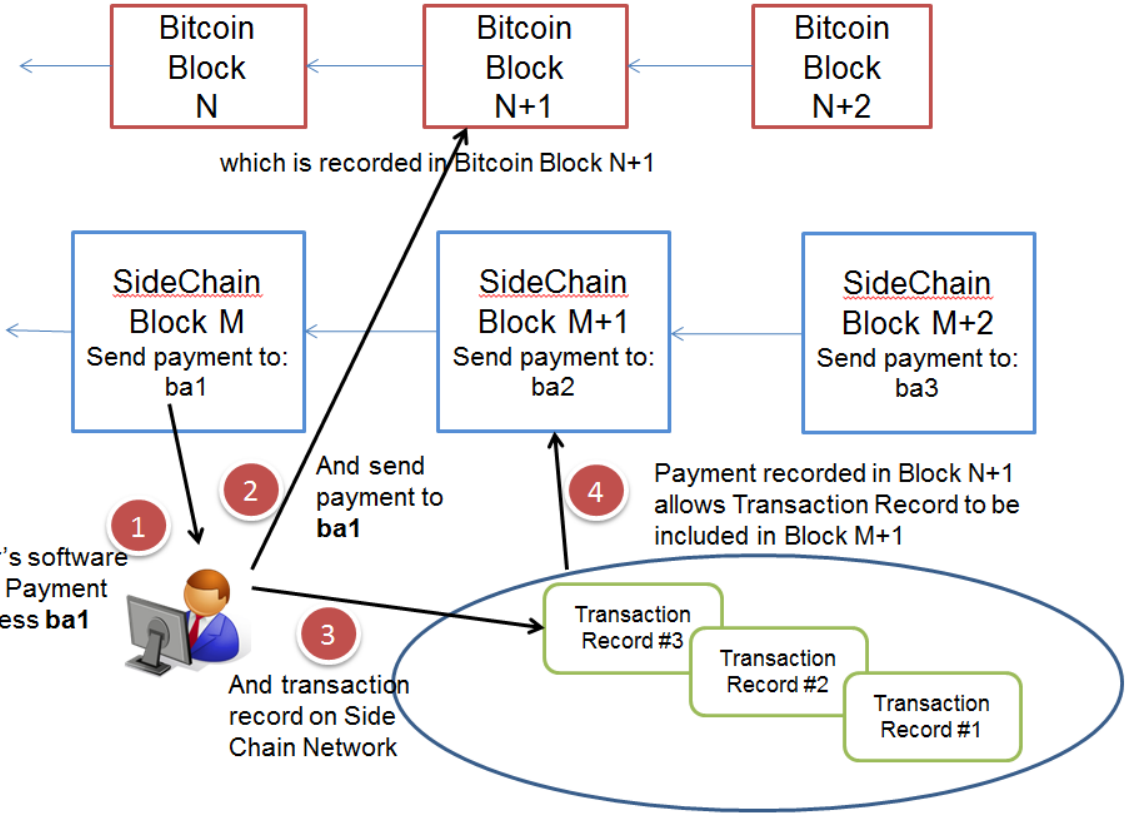

Bitcoin Block N
Bitcoin Block N+1
Bitcoin Block N+2
which is recorded in Bitcoin Block N+1
SideChain Block M Send payment to: ba1
SideChain Block M+1 Send payment to: ba2
SideChain Block M+2 Send payment to: ba3
1
User's software grab Payment address **ba1**
2
And send payment to **ba1**
3
And transaction record on Side Chain Network
4
Payment recorded in Block N+1 allows Transaction Record to be included in Block M+1
Transaction Record #3
Transaction Record #2
Transaction Record #1

Figure 1

But there is another significant part of the Bitcoin network that we have not yet addressed. One feature of the Bitcoin network is that it gives anyone the ability to access any of the prior blocks making up the Bitcoin block chain, of which every node has a copy. In order to perform their operations, these nodes are obligated to retain a copy of the block chain at all time in order to ensure that other nodes have a proper copy of the block chain. Doing so allows them to approve newly created blocks by verifying that all new transactions are in accordance with prior recorded transactions. However, our proposal up to this point has described no incentive for nodes to keep copies of prior side chain blocks, unless of course this side chain were to handle transactions using either its own currency or with transferred bitcoins. As with the Bitcoin network that requires nodes to share copies of prior blocks to ensure that their future blocks can be approved, so must the same principle apply to our hypothetical side chain. But then the question becomes, how?

Creating an incentive to retain all prior blocks

When a new side chain block is created, the side chain nodes would be required to run the hash algorithm that would take, as input, the contents of a prior, randomly selected side-chain block other than the immediately preceding one and as well as of the preceding block. A side-chain block header would contain the following information:

1. The Bitcoin address to which this node wants payment sent as compensation.
2. The hash of the previous side chain block.
3. The block number of the latest Bitcoin block.
4. The hash created using the contents of a prior side-chain block other than the immediately preceding one (see formula below) combined with this side chain block's number and the hash of the latest Bitcoin main block chain. (It might also include the whole block chain of the winner.)
5. The list of transactions to be included in the block.
6. This side chain hash.

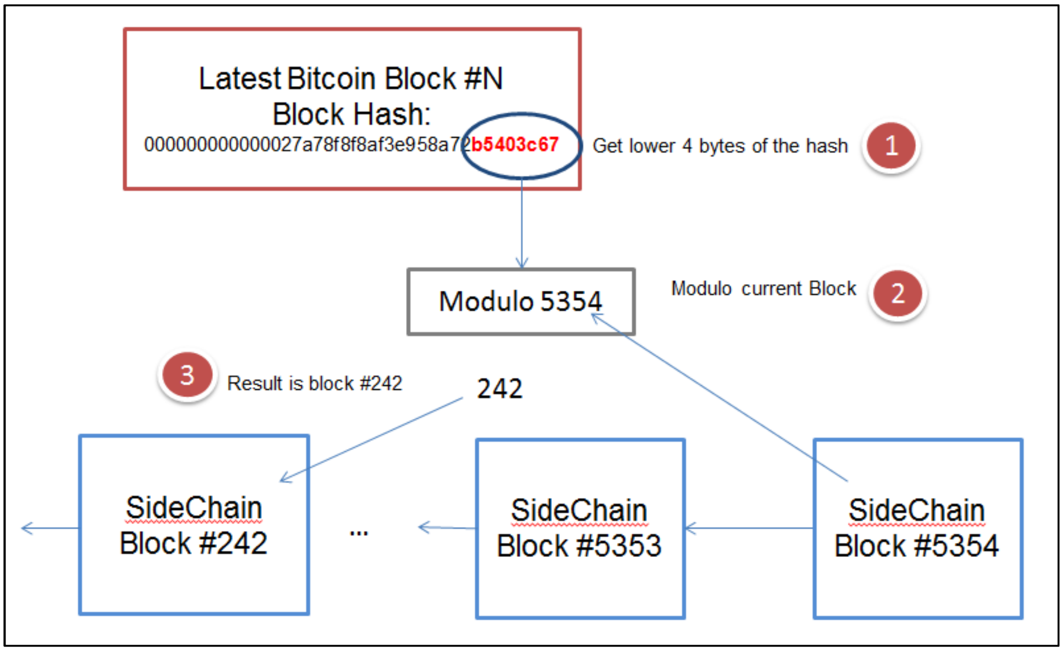


Figure 2

The hash created for Item 4 above would require the full contents of a previous side chain block unknown by all nodes until now. The selection formula would require the hash of the latest Bitcoin main block. The last four bytes of this hash (those on the left side and so least significant in value) would be used as the input number.

The formula by which the prior side-chain block number P is randomly selected is as follow:

$$P = H \text{ modulo } C$$

where

H = The last four bytes of the latest Bitcoin block hash output.

C = The number of the current side-chain block number

The use of the hash from the latest block of the main Bitcoin block chain would provide the required randomness.

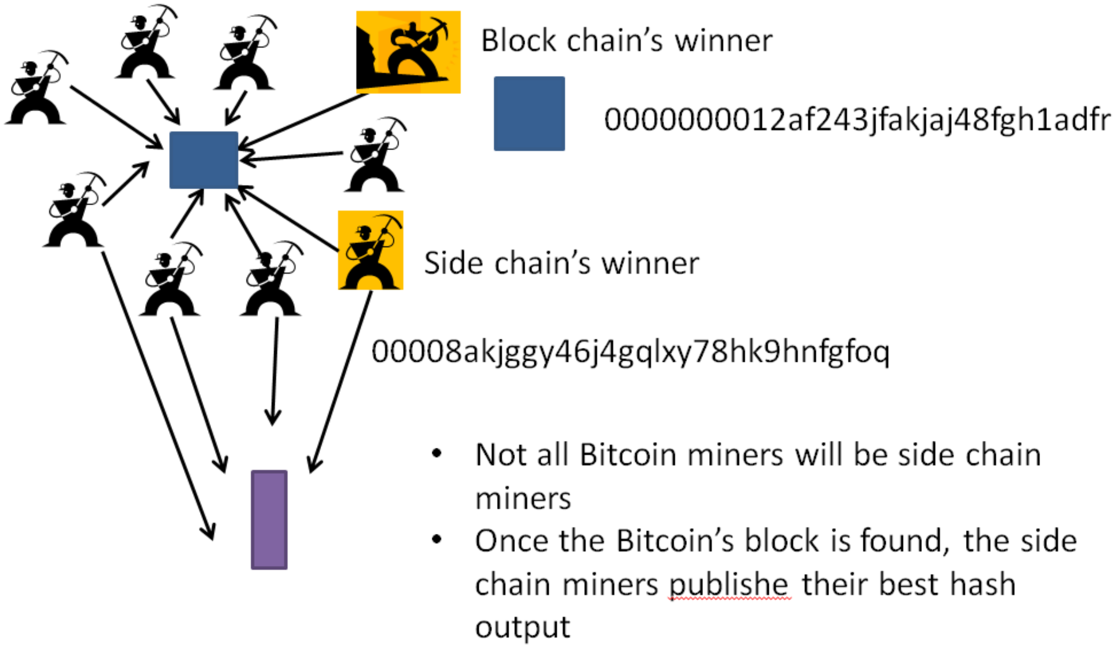
In the example shown in Figure 2, item 4 listed above would use, as input to a hash algorithm (SHA-256, for example), the full contents of side-chain block #242 combined with current block number 5354 and the hash of the main Bitcoin block.

Note that, unlike the current Bitcoin protocol that requires the full block chain to be available forever, a specific side chain might have a lighter requirement regarding the length of time that nodes must retain blocks, one year only, for example. In this case, the formula would have to be modified slightly. For instance, assume that only the last 1000 blocks are required to be kept by nodes; the formula above would then be re-stated as follow:

$$P = C - 1000 + (H \text{ modulo } 1000)$$

Determining the winning node

Existing Bitcoin nodes willing to participate as operators in this side chain would have to run the extra software needed to operate it. As they are mining the current block on the Bitcoin network, these participating side chain nodes would take note of the lowest hash value (along with its corresponding nonce) that they have obtained so far while racing to mine the current Bitcoin block. Not all miners on the main Bitcoin network might decide to participate in this side chain network, and, in that case, the best runner-up in the hash race on the Bitcoin network might be selected. When a block satisfying the current difficulty level has been published on the main Bitcoin network, these participating miners would publish their respective blocks that would have been selected as the latest block of the Bitcoin block chain had their lowest hash value met the current Bitcoin protocol’s difficulty level. The miner from among these who published the Bitcoin block with the lowest hash would become the winner selected to produce the next side-chain block; that side chain block would contain the Bitcoin address of this miner. Of course, if the publisher of the latest Bitcoin block were among the participants on this side chain, he would be the winner of the side chain as well and so would collect the transaction fees for the next block of the side chain.


Block chain's winner
0000000012af243jfakjaj48fgh1adfr
Side chain's winner
00008akjggy46j4gqlxy78hk9hnfgfoq

- Not all Bitcoin miners will be side chain miners
- Once the Bitcoin's block is found, the side chain miners publishe their best hash output

Figure 3

Now this winner would begin receiving payments to his Bitcoin address by current users of the side chain network who have their transactions recorded and included in the next side chain block.

Incorporating side chain transactions

To have its transaction processed in the side chain, the user would have to send a certain amount of bitcoins – using the Bitcoin network – to the Bitcoin address listed in the latest side chain. The user would "sign" his transaction record with the private key corresponding to the bitcoin address used to send the bitcoins to the published Bitcoin address, thus confirming that payment has been made for this transaction record on the side chain. As explained earlier, miners (or side chain "operators") would be paid from the transaction record fees of the next side chain block following theirs.

Splits and orphan blocks on the side chain should not occur since the lowest hash value would always be selected. As opposed to the current Bitcoin protocol which only require that a block meet a certain threshold (i.e., have a hash lower than a certain value) to be accepted, the side chain in this case would pick the hash with the lowest value. Thus, even if two miners on Bitcoin, who are both also side chain operators, happen to discover the block solution at the same time and generate a split on the Bitcoin network, their having the same hash value would be almost impossible. One of them would have a lower value and is the one picked for the side chain.

Case study #1: A block every 24 hour

Let’s assume we want to create a side chain that only requires that a new block be created every 24 hours as shown in Figure 3 below and that this block records public documents or contracts. Further assume that the winner of the first Bitcoin block created after midnight GMT has been selected to create this block. The Bitcoin main-chain miners who competed to obtain the hash output satisfying the difficulty level kept a record of their best (i.e., lowest value) hash output results along with their corresponding nonce. They then broadcast just their hash output to the other operators on the network. If required, a random delay of 1 to 30 seconds could be implemented so that not all nodes are flooding the network with packets at the same time. If a node receives a hash output having a lower value than his, he need not bother sending his own.

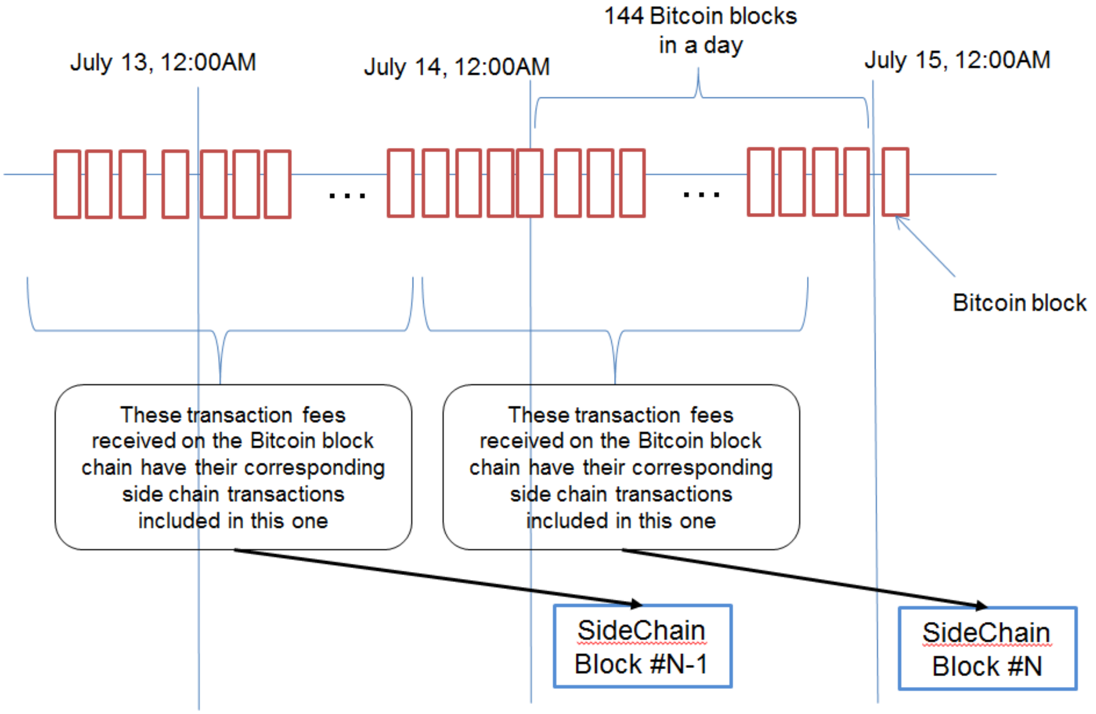

144 Bitcoin blocks in a day
July 13, 12:00AM
July 14, 12:00AM
July 15, 12:00AM
Bitcoin block
These transaction fees received on the Bitcoin block chain have their corresponding side chain transactions included in this one
These transaction fees received on the Bitcoin block chain have their corresponding side chain transactions included in this one
SideChain Block #N-1
SideChain Block #N

Figure 4

Once the winning (i.e., runner-up) node has been identified, it starts to work on creating the side chain block incorporating all transaction records received during the last 24 hours. Since a new block is created only once every 24 hours, this side chain has the luxury of picking transactions for which transactions fees have been recorded on the main block chain in more than 6 blocks, securing a comfortable confirmation. As illustrated in the figure, there can be a delay of 6 blocks, hence the operator picks up the transactions with their corresponding transaction fees recorded in the last 6 blocks prior to midnight the day before and stop 6 blocks prior to this midnight.

An immediate concern arises when an operator simply decides not to include any side chain transactions within the side chain block he creates since transaction fees are detached from the inclusion process. Thus, users who have paid transaction fees on the Bitcoin block chain would not have their side chain transactions included in the next side chain block. Karma-like, the next users would most likely not send any bitcoins to the Bitcoin address of this latest operator, who deliberately excluded transactions, and the next miners would pick up those previous transactions not included in a previous side chain block, and consequently, collect future transaction fees.

Case study #2: A block every 10 minutes

A 24-hour period would allow plenty of time for payment of transaction fees to be confirmed. Not so in the case of a 10-minute block period.

Here, we propose another possible alternative where miners perform the work first. The hypothetical side chain block could have two modes, *soft* and *firm*. Soft blocks would contain side chain transactions with corresponding transaction fees that are in the latest 10 blocks of the block chain. Once the payment of transaction fees on the block chain have been secured in at least 10 blocks, the corresponding side chain becomes “firm.” A soft block might have a side chain transaction removed from it if its corresponding transaction fee is determined to have not been included after 10 blocks or fewer have passed. Because of the short time, a side chain transaction would be accepted if the transaction fees have been paid to either the prior or current operator’s Bitcoin address.

Case study #3: A block every minute

Another side chain implementation could conceivably have a shorter block period than the Bitcoin protocol. In this case, the winning miner would be responsible for the next 10 or perhaps 20, 30, or 60 blocks. This situation is comparable to splitting the block into multiple blocks over an established time period. For example, if the period selected is every minute, the same operator would be selected until the solution for the next Bitcoin block was found, at which time the identity of the next operator to take over the operation of creating one block exactly every minute is established.

Conclusion

The contents of this proposal would allow a side chain to be launched immediately without the the Bitcoin protocol having to be modified. Eventually, the Bitcoin protocol will most likely be updated to include the concept of a side chain, but, until then, the method described here could be used as a way to begin experimentation with the concept. Multiple variants are possible, and I hope this document might be helpful in generating other possibly more elaborate ideas.

Extra:

Note 1:

Another twist might be to have side chain miners doing their own proof-of-work on the side chain block, potentially using merged mining. Miners would be competing for the proof-of-work with the hash native to the side chain block. Just as with the Bitcoin protocol, the hash of the side chain block would be required to fall below a certain level which is adjusted based on the rapidity of calculation by the overall network. The downside of this proposal is that extra hash power would need to be dedicated to this side chain if it is not implemented with merged mining.

Note 2:

Originally, instead of having the miner collect the reward belonging to the future block the miner could collect the reward (transaction fees) associated with the transactions he includes in the current block he is creating. However, since transaction fees are collected at the time of transmission, this would require sending it to an established Bitcoin address, possibly a multisig address of 6 out of 10 miners (the winner and 9 runners up). These miners would be co-signing multiple transactions, 1 to the winner and a smaller amount to each of the 9 runners up. But the method where the operator collects the transaction fees for a future block are not likely to bring any issues, hence no need to complicate using multisig.