Public versus Private Blockchains

Part 2: Permissionless Blockchains

White Paper

BitFury Group
in collaboration with Jeff Garzik (jeff@bloq.com)

Oct 20, 2015 (Version 1.0)

Abstract

Blockchain-based solutions are one of the major areas of research for financial institutions and in other applications across the globe. There is currently an ongoing debate whether the existing blockchain-based systems (such as Bitcoin and other cryptocurrencies) can be utilized as is in proprietary contexts, and whether their openness and censorship resistance are fitting properties in this case. We provide arguments for the use of permissionless blockchains and open blockchain protocols in creating ledgers and registries, devoting particular attention to the Bitcoin blockchain as the most commercially successful and secure permissionless blockchain. We study potential applications of permissionless chains in proprietary environments, such as colored coins, peer-to-peer payment channels and transaction processing by known validators.

Version History

Version	Date	Change description
1.0	Oct 20, 2015	Initial version

Bitcoin [1] is a peer-to-peer digital currency system addressing the problems of inadequately slow speed of centrally processed financial transactions. Unlike centralized databases used in modern financial environments, Bitcoin relies on a distributed ledger called the *blockchain*, with transactions grouped into cryptographically secured blocks. Compared to traditional financial solutions, Bitcoin differs in three key aspects:

- **Decentralization.** The trust in the Bitcoin blockchain does not arise from an authority maintaining the ledger, but rather from the mathematical soundness of the system and a prohibitive economic cost of an attack. As such, each Bitcoin user can independently verify the current state of the ledger and arrive to the same conclusion as the rest of the network.
- **Resistance to censorship.** The Bitcoin blockchain is permissionless – any user can broadcast any transaction conforming to the Bitcoin protocol to be included into the blockchain. Similarly, any user can participate in updating the blockchain (*mining*).
- **Append-only.** The major design goal of the blockchain is to make it such that deleting information from the blockchain (i.e., reversing transactions) is impossible or at least prohibitively expensive.

The idea of a blockchain is a core aspect of Bitcoin; it has been recreated with varying degrees of success in many digital currencies that emerged afterwards, such as Ripple [2], Litecoin [3], Ethereum [4], BitShares [5], Nxt [6], and others. Blockchains in these currencies differ from the one used in Bitcoin in specifics of the protocol (e.g., methods of securing blocks of transactions), whereas the core design features of the Bitcoin blockchain mostly remain intact.

Institutions operating data registries and financial ledgers are cautious to use publicly available blockchain solutions. We examine oft-repeated reasons behind the hesitation to use permissionless blockchains in finances – for example, concerns about transaction processors' identities, theoretical lack of transaction finality, and susceptibility to attacks in Section 1. In Section 2, we review solutions that could leverage permissionless blockchains for registry and ledger applications. Finally, in Section 3, we consider benefits of permissionless blockchains and standardized open blockchain protocols; we contend that permissionless chains could create the ubiquitous peer-to-peer trustless layer for blockchain innovations. See the first part of the paper for an introduction to blockchain technology, a review of the present state of implementing blockchain innovations and a description of permissioned blockchains.

1 Perceived Problems of Permissionless Blockchains

Blockchain technology provides a distributed, append-only, fault-tolerant timestamped registry that could be used e.g. as a ledger in financial applications. However, there are several considerations that make available public permissionless blockchains problematic from the point of view of registry and ledger applications. We examine these problems focusing on Bitcoin as the most commercially successful public blockchain; similar concerns pertain to other cryptocurrency blockchains.

1.1 Mining Process

Problem. Bitcoin is a permissionless blockchain; a block can be created by any node of the network, provided it has enough computational power to satisfy block header validity criteria (proof of work). On the other hand, according to many jurisdictions, financial transaction processors must have an established identity. In theory, the anonymity of bitcoin miners makes it problematic to consider Bitcoin as a proper medium for processing transactions.

In practice, starting in 2013, mining has been performed using application-specific integrated circuits (ASICs); there are a relatively small number of active independent miners at any given moment of time, as individual miners are largely aggregated into pools. The identity of a block miner can be determined based on the properties of the coinbase transaction; these recovered identities are commonly displayed on Bitcoin explorer websites [7, 8]. To ensure rigorous miner identification, a miner can include a digital signature of block contents into the coinbase transaction; see Appendix A for technical details.

If an institution wants to ensure that related Bitcoin transactions are mined by accredited miners, it may send transactions over a secure channel directly to these miners rather than broadcasting them over the network; accepting non-broadcast transactions into blocks is a valid behavior according to the Bitcoin protocol. See Section 2.1 for more details.

1.2 Privacy Concerns

While cooperation between miners and financial institutions could solve the problem of undesired publication of unconfirmed transactions, it leaves unresolved the problem of clients' confidentiality.

Problem. As the whole transaction history in Bitcoin is public, clients with transactions on the Bitcoin blockchain may have concerns that their transaction history can be used in unlawful ways.

Bitcoin does not have user identification; the system uses public key security to identify the ownership of bitcoins. However, a user utilizing the same public key (or, equivalently, the same Bitcoin *address*) for multiple payments could be identified by his payment history. This problem can be mitigated using hierarchical deterministic (HD) wallets [9]. This type of wallet is able to algorithmically generate a new public key for every transaction. More than that, due to the homomorphic property of the elliptic curve cryptography used in Bitcoin, the knowledge of private keys is not required; this makes HD wallets a secure solution for large financial firms. To an outside observer, there is no easy way to link together the addresses generated by the wallet. A more complicated example utilizing homomorphic property of elliptic curves is the pay-to-contract protocol [10], enabling one-time addresses for payments that depend on the payment specification (e.g., quantities and types of ordered goods).

A private blockchain environment, especially if dealing with smart contracts, can utilize secret sharing schemes [11]. This goal is currently being pursued by the Enigma project [12]. In the case of a Bitcoin-like ledger using unspent transaction outputs (UTXOs), the secrets would be values associated with outputs (i.e., an outside observer would not be able to determine how much transaction value is

being moved). Nodes of the network would have no access to these values but only to their encrypted presentations. The goal of the nodes would be to verify whether the values of created outputs satisfy certain conditions:

- each output value is non-negative
- the sum of the output values is no greater than (or, if there is no transaction fee, strictly equal to) the sum of outputs spent by the transaction.

The systems assuring evaluation of these conditions have already been implemented with the help of shared secrets (see, e.g. [13]). In the blockchain context, encrypted UTXO values satisfying the above conditions have been implemented within Sidechain Elements by Blockstream [14].

1.3 Cryptocurrencies as Bearer Assets

From the legal point of view, Bitcoin and other cryptocurrencies could be classified as bearer assets: no ownership information is recorded and the assets are issued in physical form to the buyers; the ownership of assets is determined by the sheer knowledge of the corresponding private key, similarly to how the owner of cash is a person that possesses it. It immediately follows from the abstract of the Bitcoin white paper that this kind of ownership was the design goal of Bitcoin: "A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution." As most securities represent registered assets (i.e., assets with ownership not solely determined by possession), one could argue that the Bitcoin blockchain is not suitable for their storage [15]. There are several considerations that make this statement not so unequivocal.

First, the transfer of digital assets is not stored by the means of the Bitcoin protocol; the protocol is unaware of digital assets and can only recognize and verify the move of value measured in bitcoins. Systems integrating digital assets with the Bitcoin blockchain utilize various colored coin protocols to encode asset issuance and transfer (see Section 2.2 for more details). There is nothing preventing such a protocol to be more adapted to registered assets. Asset transactions can use bitcoins as a universal currency, similarly to how "real-life" securities can theoretically be bought or sold for fiat currency. Even if an asset transaction does not involve Bitcoin directly, the Bitcoin blockchain is utilized as a secure, public, append-only store of information allowing it to reliably timestamp transactions. There are valid concerns regarding increased attack risks caused by storing high-value assets on the Bitcoin blockchain; we consider them in Section 1.6.

Second, multisignature schemes [16] allow for the creation of limited trust in the Bitcoin environment, which can be beneficial when dealing with registered assets and in other related use cases. Whereas raw bitcoins are similar to cash, multisignature schemes act not unlike debit cards or debit bank accounts; the user still has a complete control of funds, and a multisignature service provides reputation and risk assessment services for transactions. For example, multisignature services could allow merchants to accept transactions immediately, as the service's signature implies a substantially

lower risk of fraudulent activity than in the case of an ordinary Bitcoin transaction. Another use case of multisignatures is detection of user activity commonly associated with hacked accounts (such as an attempt to transfer all funds to another address); the service could detect this activity and refuse to sign corresponding transactions or require some sort of confirmation (e.g., by phone).

One of the use cases of the 2-of-3 multisignature scheme is escrow involving a mediator trusted by both parties. A buyer purchasing certain goods locks his cryptocurrency funds with a multisignature lock, which requests two of the three signatures: the buyer's, the seller's, and the mediator's. Thus, the funds can be redeemed

- by the seller if he reaches an agreement with the buyer (a successful trade)
- by the buyer if the buyer provides evidence to the mediator necessitating a refund
- by the seller if evidence is provided to the mediator that the exchange of goods has been performed duly despite the lack of agreement between the buyer and the seller.

Third, the overall trend of decentralization favors Bitcoin-like systems, for which the regulation is rigorously expressed in the protocol rules rather than dictated by a single entity. From the legal point of view, public key cryptography used in Bitcoin is not less secure than website authorization systems, which can be legally binding.

1.4 Native Tokens in Permissionless Blockchains

Problem. Institutions may be wary of blockchains with native tokens such as bitcoins. Instead, institutions may prefer a blockchain design with no native tokens (e.g., with transactions representing transfer of fiat currency or assets).

Native tokens in permissionless blockchains are required to provide the economic and game theory incentives than would compel users to maintain and grow the system. For example, newly mined bitcoins in each block and transaction fees provide an incentive for bitcoin miners to maintain security of the Bitcoin blockchain in the form of the network hash rate. In effect, the amount of rewards for creating new blocks is a good measure of security of a permissionless blockchain.

The similar reward protocol could be organically introduced into a permissioned blockchain environment if transaction processing and securing block headers are decoupled, e.g. with merged mining. In this case, a transaction similar to a coinbase transaction would be included into each block, which would give a reward to the entity securing the block header. Note that the the rewards on a permissioned chain would not necessitate an introduction of a native blockchain token, but would rather conform to the format of other transactions on the blockchain. The reward protocol described above would be advantageous compared to off-chain contracts with security providers, as it is independently auditable and is enforced algorithmically, therefore aligning with the spirit of blockchain technology.

1.5 Transaction Finality

Problem. No transaction stored on the Bitcoin blockchain can be considered truly final. Any transaction can theoretically be deleted from the blockchain by reorganizing it starting from the block containing the transaction in question.

At first glance, the problem is exacerbated by regularly observed reorganizations of the Bitcoin blockchain resulting from network inefficiency. These reorganizations occur because of the non-negligible time needed to propagate a block to all nodes of the network. If two nodes of the network discover a new block at the same time, according to the Bitcoin protocol, the network will be split into two parts each accepting the respective discovered block as valid. This type of split is quickly reconciled and is statistically unlikely to last for more than a couple of blocks. Thus, benign splits do not permanently delete transactions from the blockchain but rather slow their confirmation. Several mechanisms are proposed to diminish the probability of network splits, e.g., invertible Bloom lookup tables [17], which allow blocks to propagate substantially faster.

An attacker may want to intentionally cause a blockchain reorganization in order to reverse a certain transaction. To fight reorganization attacks and benign blockchain reorganizations, a transaction can be considered *practically final* when it has enough confirmations. (The number of confirmations for a transaction is the number of blocks on the blockchain including the block containing the transaction and all later blocks. For transactions not yet included into a block, the number of confirmations is 0.) The longest reorganization of the Bitcoin blockchain occurred in 2013 and involved 24 blocks; it was related to a bug in the protocol, and not malicious activity [18]. Similarly, two comparatively long reorganizations (6 and 3 blocks), which took place in July 2015, were caused by an improper mining procedure [19]. Thus, transactions with at least 36 confirmations (corresponding to 6 hours transaction age under normal conditions) could be viewed as practically final.

Consider the case where a transaction is considered practically final if it has at least H confirmations. The goal of the attacker is to make a transaction practically final and then reverse it. A hypothetical attack scenario could occur as follows:

1. An attacker sends a transaction **Tx** that he wants to reverse on the network, or spots it in a block.
2. The attacker starts to build an alternative blockchain based on the block prior to the block containing the transaction **Tx**. The attacker needs to act in secrecy: if the attacker's chain is published before **Tx** gains H confirmations, the attack becomes obvious to the counterparty.
3. After **Tx** gains H confirmations and the attacker's chain is longer than the "valid" one, the attacker publishes his chain whole.

Note that if **Tx** does not spend the attacker's funds, it returns to the pool of unconfirmed transactions after the attack is complete and can be confirmed in the future.

In order to accomplish this type of attack with 100% probability, the attacker needs to control more than 50% of the network hash rate for the period of the attack [20]. The attacker can either buy the corresponding amount of hashing equipment or bribe existing miners into participating in the attack. In

the first case, preparations for the attack would most probably be noticed. In the second case, provided the attack needs to take comparatively long time (e.g., if 36 necessary confirmations would be needed), it would likely be detected before it is finished as flagged by the significantly increased duration between blocks. As there is a comparatively small number of mining pools, those pools participating in the attack would be detected as well. In any case, the attack is obvious after it would be performed, as lengthy reorganizations of the blockchain are statistically highly unlikely.

In the current version of the Bitcoin protocol, once the attacker's chain is published, it *would* take over even if all participants of the network are aware of the attack. To prevent this, a protocol could be modified to reject reorganizations lasting more than a specified number of blocks (as it is done in Nxt). However, this would make the Bitcoin protocol weakly subjective [21], introducing a social-driven security component into the Bitcoin ecosystem. On the other hand, if the change is implemented, it would introduce truly final transactions. For example, if reorganizations of more than 35 last blocks are not tolerated, transactions with 36 or more confirmations are final and cannot be modified.

1.6 Censorship Attack

Problem. Consider a malicious entity, which accumulated more than 50% of the hashing power of the Bitcoin network. Such an entity would be able to censor arbitrary transactions by not including them into blocks mined by the entity and by rejecting all blocks containing censored transactions. Although a long-lasting attack of such kind would not be profitable for the attacker, the attack could be perpetrated for other reasons (e.g., to demand ransom).

Censorship resistance is the defining feature of Bitcoin; negating it by the described attack would result in a negative effect on both the price and reputation of the currency. To prevent censorship attacks, the Bitcoin protocol would need to detect and prevent censorship behavior, which is manifested in two ways:

- the attacker ignores blocks mined by the honest part of the network
- the attacker does not include certain transactions in his blocks.

One of the possible approaches to the problem would be compulsory synchronization of the pool of unconfirmed transactions (the logic would be as follows: if a transaction in the pool for unconfirmed transactions is sufficiently old and if its priority determined by the transaction fee is sufficiently high, it needs to be included in the pool for all mining nodes). As synchronizing unconfirmed transactions would increase the propagation speed of mined blocks, synchronization logic is included in the invertible Bloom lookup table proposal [17].

The security of the Bitcoin network in the case of economic equilibrium is determined by the rewards received by block miners and is therefore tied to the exchange rate of Bitcoin. Thus, creating high transaction throughput of expensive digital assets on the Bitcoin blockchain with the help of colored coin protocols has certain risks: it increases the potential gain from an attack on the network, while security of the network could remain roughly the same (as there are no specific fees for digital

asset transactions; transaction fees for these transactions are still paid in bitcoins). The risk can be mitigated if Bitcoin fees for asset transactions would be consciously set high, either by senders or by a colored coins protocol itself, allowing Bitcoin miners to improve security of the network according to the value transferred both in bitcoins and in digital assets.

2 Permissionless Blockchain Applications

As in Section 1, we concentrate on Bitcoin. The Bitcoin blockchain is presently the most secure existing permissionless chain in terms of attack costs; in the permissionless environment, the cost of an attack is proportional to mining rewards, which in the case of Bitcoin measure to ≈$900,000 per day. At the same time, the price of maintaining security is relatively low for Bitcoin users, as it can be measured by two factors:

- transaction fees (about $2 \cdot 10^{-4}$ bitcoins per transaction on average at the time of writing, i.e., approximately $0.05)
- controlled inflation subsidy (approximately 9% per year as of 2015).

Thus, the Bitcoin blockchain is a primary choice among existing permissionless chains for rational economic actors.

2.1 Known Transaction Validators

As there is a relatively small number of Bitcoin mining pools, miners can act as known processors of Bitcoin transactions originating from institutions (e.g., due to compliance reasons). The cooperation with institutions could take the form of encrypted channels for Bitcoin transactions established between institutions and miners. For example, transactions sent by institutions could be encrypted using an asymmetric encryption scheme such as ECIES [22] with the miner's public key; transactions encrypted in this way can only be decrypted by the miner. It is important to stress that the proposed cooperation does not imply censorship of Bitcoin transactions created by ordinary Bitcoin users.

Currently, Bitcoin has no agreed upon tools to accept a transaction into a pool of unconfirmed transactions without broadcasting it to the network. However, this functionality could easily be achieved by refactoring the **sendrawtransaction** RPC method of Bitcoin Core [23], as it first checks the validity of a transaction and accepts it to the local transaction pool if the transaction is valid (which is the exact functionality required) and then relays the transaction over the network. If needed, transactions received from financial institutions would be prioritized to speed up their inclusion into a block using the **prioritisetransaction** RPC method [24]. In the ideal case though, these transactions would be prioritized solely based on their transaction fees (i.e., in a same way *all* Bitcoin transactions are prioritized), which at the same time would constitute payments for the validation by a known entity. Thus, this form of transaction processing would align with the core assumption for Bitcoin mining

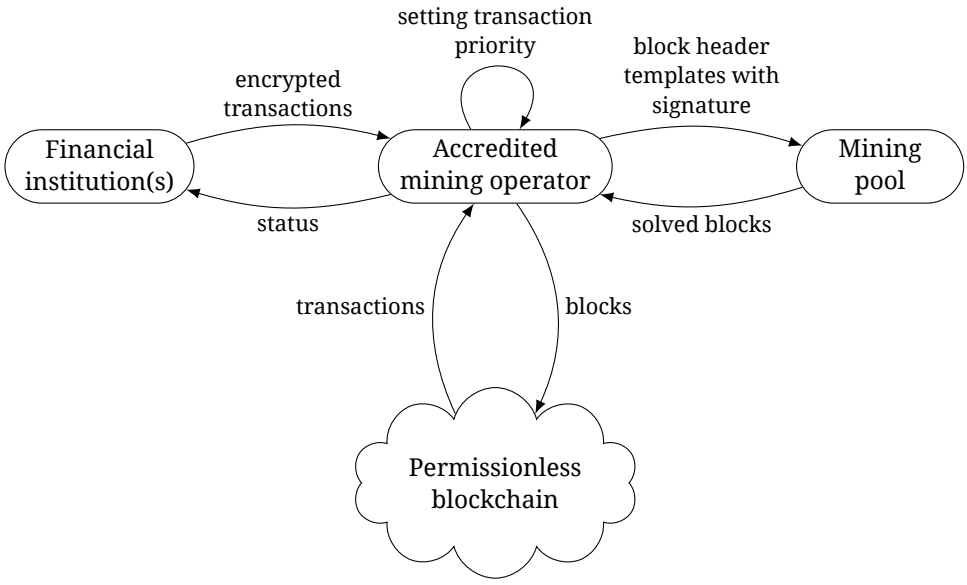

Figure 1: Secure channels between financial institutions and Bitcoin miners to place transactions on the Bitcoin blockchain. Bitcoin miners prove their identity by signing blocks with digital signatures as described in Appendix A

that miners are rational economic actors and try to maximize their profit. Note that inclusion of non-broadcast transactions into blocks could diminish the effectiveness of certain scalability efforts such as invertible Bloom lookup tables described in Section 1.5.

Additionally, partnerships between institutions and miners minimize risk in case transactions should not be made public before they are confirmed. Similar to how premature disclosure of information about large-volume share trades can have negative consequences for businesses, information about unconfirmed financial-related transactions could act as a negative factor in trading. Partnerships with Bitcoin miners are effective even if their mining power is geographically distributed (e.g., for mining pools); mining protocols used in distributed mining such as Stratum [25] already restrain from disseminating transactions among participants of the mining pool. Thus, only the pool administrator needs to have access to transactions; the ordinary participants of the pool do not have access to them and cannot recover transactions from the information provided for mining.

2.2 Colored Coins

Colored coin protocols are a primary means to embed user-defined digital assets into the Bitcoin blockchain. These protocols use small pieces of data (typically no longer than 40 bytes) to encode issuance and / or transfer of digital assets into ordinary Bitcoin transactions. The protocols (Table 1) differ as to how the data is encoded and in whether or not additional middleware layer is required for performing digital asset transactions.

- Protocols such as Open Assets Protocol, Colored Coins Protocol, and ChromaWay provide a readily available framework for asset transactions; these protocols do not rely on any additional software.

- More complicated protocols (Counterparty, OmniLayer, CoinSpark) require specialized or complemented versions of Bitcoin node software providing a richer environment for asset transactions (e.g., automatic order matching, dividend payments, and so on). The middleware layer is usually open sourced and is over a distributed network (similarly to Bitcoin itself) so that it does not constitute a single point of failure.

Table 1: Colored coin protocols

Name	Website	Year of foundation	Middleware required
ChromaWay	chromaway.com	2012	no
Open Assets Protocol	github.com/OpenAssets	2013	no
Colored Coins Protocol	coloredcoins.org	2015	no
OmniLayer / Mastercoin	omnilayer.org	2013	yes
CoinSpark	coinspark.org	2014	yes
Counterparty	counterparty.io	2014	yes

Services that build atop of colored coin protocols include asset transaction explorers, colored coin wallets, and cloud platforms (Storj and MaidSafe). One of the interesting financial applications of colored coins is Tether (tether.to), a service using colored coins to represent US dollars for fast money transfer. Several cryptocurrencies such as Nxt and BitShares support custom digital assets natively.

Compared to permissioned blockchains, colored coins technology has certain advantages for use in financial applications (Fig. 2):

- Colored coins rely on existing infrastructure; the solutions based on this technology would require less development and maintenance efforts
- Colored coins are more transparent for participants and auditors compared to permissioned blockchains
- As colored coins operate on top of permissionless blockchains, systems using colored coins are inherently resistant to censorship – restrictions on transactions are fully specified by a colored coins protocol instead of being enforced by a certain entity
- Existing colored coins implementations are open source, which allows users to perform the independent audit of the code to ensure the system possesses necessary properties.

On the other hand, permissioned blockchains are more tunable to financial needs:

- Permissioned blockchains provide balance between user confidentiality and transparency (e.g., they can grant differing types of access permissions to blockchain data for various categories of users)

- There is less overhead compared to colored coins protocols. Transaction verification for permissioned blockchains can be finely tuned for specific types of assets. A permissioned blockchain may include complex scripting tools enabling smart contracts, whereas colored coins are mostly limited to value transfer.
- Permissioned blockchains may be more compliant (e.g., due to known identities of transaction processors).

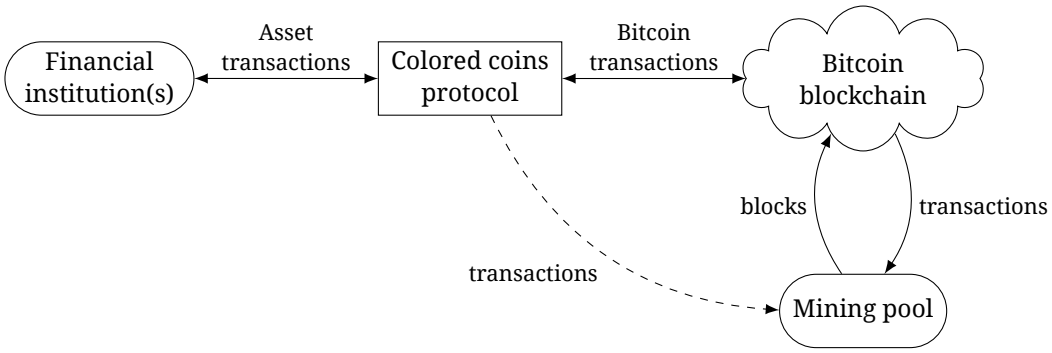

Figure 2: Using colored coins on top of the Bitcoin blockchain to implement asset transactions. For compliance, financial institutions may use secure communication channels with miners described in Section 2.1 to place asset transactions on the blockchain

Note that financial institutions could implement proprietary colored coin protocols reflecting their needs:

- more restricted set of rules concerning ownership and transfer of digital assets, e.g. for compliance reasons
- better representation of semantics of concrete types of assets, e.g. limited duration
- encryption of data for client confidentiality.

In general, a weighted approach using both colored coins and permissioned blockchains can optimally combine the respective benefits of each.

2.3 Sidechains

Pegged sidechains [26] allow for integration of multiple financial blockchains into a single interconnected environment, as well as their integration with permissionless blockchains. The goal of sidechains is to allow funds or asset transfer from one blockchain to another at a fixed or otherwise deterministic exchange rate. The fully trustless implementation of the sidechain concept would require for protocols of both blockchains to be aware of other chains; to our knowledge there are currently no public blockchains implementing this requirement. Transfer between blockchains can otherwise

be achieved using independent oracle services that verify cross-chain transfers (*federated peg*) or exchange between users (*atomic swaps* [27]). These kinds of transfers could straightforwardly be implemented in Bitcoin.

The generic idea behind sidechains is the following:

1. A user who wants to transfer assets from Blockchain A to Blockchain B moves assets on Blockchain A to a specially crafted locked output.
2. The user then creates a transaction on Blockchain B referencing the locked output. The protocol on Blockchain B should be able to verify that the assets are present and locked on Blockchain A using simplified payment verification.
3. The user proceeds to transfer assets, now on Blockchain B; for example, those assets can be sold to another user. If an owner wants to transfer them back to Blockchain A, he sends them to the locked output on Blockchain B; referencing this output allows to unlock the corresponding output on Blockchain A.

In the case of using a federated peg method in Bitcoin, locking and unlocking can achieved through pay-to-contract addresses [10], which utilize the homomorphic property of elliptic curve cryptography. Pay-to-contract addresses are created anew for each transfer from one blockchain to another based on a multisignature template which necessitates signatures of multiple oracle services (e.g., 4 of 5). While oracles' public keys are known, the signatures to unlock the output need to be produced with *modified* oracles' private keys, with the modification dependent on the details of a payment. The homomorphic property of elliptic curve cryptography enables creating pay-to-contract addresses from a multisignature template by modifying oracles' public keys in the same way their private keys need to be modified (i.e., a user is not required to cooperate with oracle services to produce such an address). As cross-chain trading addresses differ for each transfer between chains, they are undetectable and are resistant to censorship.

The sidechains concept could be used by financial institutions to provide an easy cross-trading mechanism among specialized blockchains of the same bank group (e.g., corresponding to different kinds of securities) or among blockchains belonging to various groups. Relatively simple Bitcoin-like blockchains used for pure financial transactions could form the basis of the financial environment, while more complex Ethereum-like blockchains would be utilized for smart contracts that move value on the base chains (Fig. 3). This approach would leverage security of Bitcoin-like blockchains with speed and versatility of smart contract chains. The open and standardized design of blockchain protocols would simplify their integration. The environment integrating smart contracts and asset transfer on top of the Bitcoin blockchain is currently being developed by Counterparty [28]; a similar goal of building a Bitcoin sidechain for smart contracts is being pursued by Rootstock [29].

Bitcoin and other public permissionless blockchains could be a part of the interconnected financial environment similarly to how cash is a ubiquitous part of the banking system. More concretely, cryptocurrencies could be used as

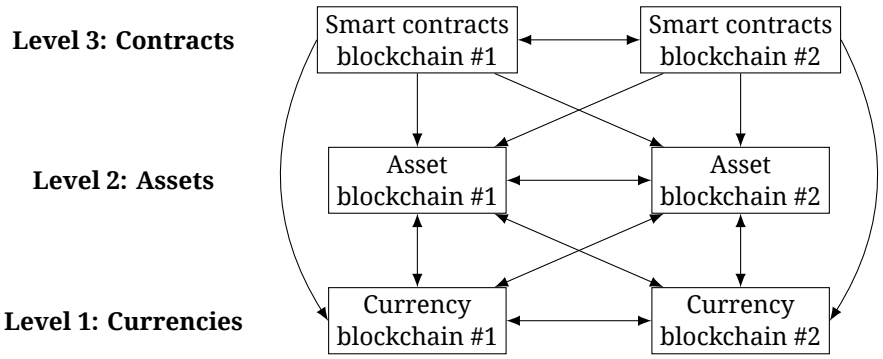

Figure 3: Multi-layered financial blockchain environment with components connected via sidechain technology

- one of the means to buy and sell assets on permissioned blockchains
- an instrument that enables relatively fast value transfer among permissioned blockchains
- an agreed upon medium for clearing operations among blockchains maintained by various institutions (Fig. 4).

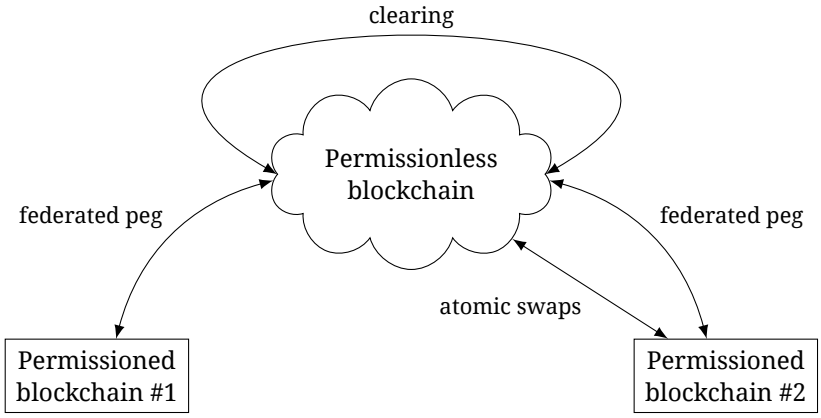

Figure 4: Using public permissionless blockchain for clearing operations among proprietary blockchains maintained by financial institutions

2.4 Peer-to-Peer Payment Channels

Peer-to-peer trustless channel networks such as Lightning Network [30] could provide an answer to scalability issues for Bitcoin and other public blockchains. The core idea behind payment channels is the following: if there exists a stable stream of blockchain-based payments between two parties (e.g., an online streaming content provider and its user), intermediate transactions do not need to be recorded on the blockchain; the only transactions that need to be recorded are the initial funding transaction and the transaction closing the payment channel. Thus, the blockchain could adapt to throughput measured in thousands of transactions per second while keeping the blockchain size in reasonable bounds; the blockchain is used as a reliable ledger to define and settle payment contracts.

The proposed design of Lightning Network is

- **decentralized** – there is no mediator required in creating and maintaining the channel
- **trustless** – the channel design is resistant to malicious activity of one of the parties or a third party.

These features necessitate the complex structure of intermediate transactions (each new payment from one party to the other induces at least 8 transactions). While payment channels with a mediator could be more simply organized, they have scalability issues as the mediator would need to process all active channels.

Lightning Network provides high degree of scalability in the form of hashed timelock contracts (HTLCs). HTLCs allow to securely conduct payments between users who are not directly linked by an open payment channel by using one or more intermediate network nodes (Fig. 1). Thus, organization of payments is similar to the Ripple network, with the important distinction that the Ripple network is based on trust, whereas Lightning Network is completely trustless.

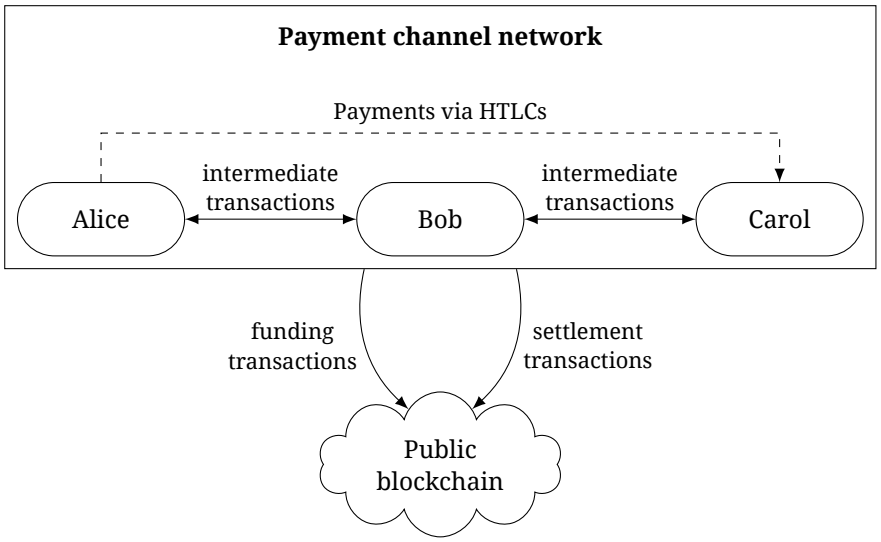

Figure 5: Organization of a payment channel network on top of the public blockchain

Payment channels based on principles of Lightning Network could be built on top of any public blockchain that supports core Bitcoin functionality (including permissioned chains described in the first part of the white paper). Payment channels could be used jointly with a colored coin protocol in order to encode transfer of digital assets instead of the native token of the blockchain.

3 Requirement for Permissionless Blockchains

While permissioned blockchains better comply with existing regulations and therefore would be more attractive for introducing blockchain databases into existing ledger and registry applications in the medium term, they limit one of the core aspects of blockchain technology – *trustlessness*. Blockchains are designed with Byzantine fault tolerance [31] in mind, i.e. they expect possible malfunctions and

dishonest behavior from the nodes of the blockchain network. On of the goals of blockchain technology is to remove human factor from transaction processing replacing it with rigorously defined and publicly available protocol, which would be enforced by a network of independent computers.

If a blockchain is completely opaque for its end users (e.g., a blockchain-based banking system that still uses legacy communication interfaces such as credit cards), the trustless aspect of blockchains is substantially reduced. End users cannot even be sure that a blockchain system is indeed in use, much less to independently verify the correctness of blockchain data (as there is no access to data and no protocol rules to check against). Human factor remains a vulnerability in private blockchain designs as long as the state of the blockchain is not solely based on its protocol, which is enforced automatically with as little human intervention as possible. Interaction based on legacy user authentication interfaces would be a major source of vulnerabilities in the case of the opaque blockchain design; new interfaces based on public key cryptography could reduce the associated risk of attacks.

Proprietary nature of private blockchains makes them less accessible; open sourced and standardized blockchain implementations would form a more attractive environment for developers and innovations. In this sense, blockchains with a public protocol are similar to open Internet standards such as IP, TCP and HTTP, while proprietary blockchain designs could be similar to proprietary Internet protocols that did not gain much traction. A proprietary blockchain protocol could contain security vulnerabilities that remain undiscovered and exploited for a long time, while a standardized open blockchain protocol could be independently studied and audited. This is especially true for protocols of permissionless blockchains, as users have a direct economic incentive to discover vulnerabilities in the system in order to exploit them. As the Bitcoin protocol has been extensively studied by cryptographers and scientists in the field, it could arguably form the basis for the standardized blockchain design.

Permissioned blockchains with regulated or public access to data would be more secure than private chains. These chains would provide access to blockchain data and the blockchain protocol, which enables independent auditing and transaction validation (although these features may be limited to regulators and end users). While clients may desire privacy, limiting access to the blockchain makes it difficult to audit, especially in the case of an interconnected blockchain environment.

The limited set of transaction validators on permissioned blockchains raises concerns:

- Authentication of validators introduces a vulnerability to the system. (Note that a single hacked validator cannot do much harm to the blockchain in the case of a reasonable blockchain protocol and proper decentralization of transaction processing.) While in the case of permissionless chains, transaction processing is inherently neutral and equidistant from all parties, permissioned blockchains have a few trusted operators. This leaves a wider possibility of corruption, tampering and systemic monitoring.
- The key design element of blockchains is “embedded economy” – a superset of embedded security and transaction validation. Each blockchain forms its own economic ecosystem; a centrally controlled blockchain is therefore a centrally controlled economy, with all that entails.

- There exists a risk of validators colluding to create multiple copies of the blockchain and / or alter chunks of it. This risk could be mitigated by introducing proof of work into the blockchain design.
- It is not clear how the blockchain would function in the case validators would become disinterested in its maintenance, or how it would recover in the case of a successful attack (cf. with permissionless blockchains, which offer the opportunity of self-organization).

Public permissionless blockchains utilize trustlessness to the full extent. Whereas private and permissioned blockchains correspond to varying levels of trust involved in interaction between end users and transaction validators, permissionless chains are built to eliminate this trust together with associated vulnerabilities. Environments with limited trust can be created within permissionless blockchains with the help of various technologies described earlier in the paper, such as multisignatures and colored coins protocols. In this case, a permissionless blockchain provides a decentralized medium for operations, similarly to how HTTP and TCP protocols form a trustless infrastructure for web services.

As permissionless systems are designed to be resistant to the malicious behavior and malfunctions, they are good for implementing peer-to-peer networks among end users. One of possible applications of these networks are scalable payment channels (Section 2.4). Layered networks, in which intermediate transactions constituting the vast majority of operations are performed in a peer-to-peer fashion among end users, whereas final settlement is reserved for established entities, could prove useful in a variety of commercial use cases such as micropayments and ubiquitous payments.

4 Conclusion

While permissioned blockchains could form the basis of blockchain innovations in the short run, public permissionless blockchains are inherently less vulnerable to attacks, as they are designed based on an assumption that parties may not trust each other. Permissionless chains minimize human factor and emphasize the algorithmic approach to security and data consistency, which are core aspects of blockchain technology. Therefore, permissonless chains could reasonably become a base layer of blockchain infrastructure, while permissioned applications could build on top of it. Additionally, permissionless environment provides necessary preconditions for developing scalable and reliable global peer-to-peer networks, e.g. for payment channels.

Existing public permissionless blockchains used in Bitcoin and other cryptocurrencies could implement certain modifications to increase their attractiveness for proprietary use cases; the potential modifications include addressing transaction finality and digital signatures of blocks of transactions. With these proposals implemented, cryptocurrencies could act as a service for fast, low-fee monetary transfers or as a medium for intra- and inter-bank operations.

Many permissioned blockchain applications could be built on top of permissionless blockchains with the help of existing technologies:

- Colored coins protocols could provide a means to encode transfer of arbitrary assets into ordinary transactions, thus extending the utility of public blockchains for financial institutions.
- Similar protocols could enable timestamping documents, assisting in development of decentralized timestamped registries.
- Payment channel networks could streamline payments by creating a scalable peer-to-peer layer on top of permissionless blockchains.
- Sidechain technology could be used to integrate permissionless and permissioned blockchains into a single interconnected environment. Sidechains could be used in cases when the protocol of the underlying permissionless blockchain is too restricting (e.g., in terms of transaction throughput or the required maximum settlement period).

The benefits of using existing public blockchains in ledger and registry applications include their transparency, as well as openness of underlying technologies and protocols. The Bitcoin environment in particular could be appropriate for use in blockchain innovations due to the following factors:

- developed infrastructure in the form of services and protocols running on top of the Bitcoin blockchain
- knowledgeable developer community
- relatively small number of mining pools with established identities, which allows them to act as transaction validators with known identities by cooperating with financial institutions
- high level of security provided by the hash rate of the Bitcoin network.

Appendix A Voluntary Digital Signatures in Coinbase Scripts

A miner who wants to rigorously prove his identity in mining a block could insert a digital signature of block contents into the coinbase script (the unlocking script of the first input of the first transaction in the block). This signature, which we call a *block signature*, may use the same elliptic curve cryptography as the transaction verification protocol. Note that as Bitcoin miners have well-established public identities, they do not necessarily need to include the corresponding public key into the coinbase script as well (cf. with signing ordinary pay-to-pubKeyHash transactions), but rather can publish it on their website or a similar secure location. Furthermore, in the present version of the Bitcoin protocol, it would be impossible to fit both a digital signature (70 bytes) and a public key ($\sim$34 bytes) into the coinbase script, as the maximum length of the latter is 100 bytes. Instead of the whole public key, the miner may insert its contracted version obtained through hashing and / or truncation.

As the block signature indirectly influences the block header through the Merkle root value, the protocol cannot demand to sign the block header directly. Instead, a miner could calculate a *modified Merkle root* by replacing the coinbase script with an zero-length array and sign a *modified header* obtained from the block header by replacing the Merkle root with the modified Merkle root. This

technique makes the signature independent of the main source of entropy in mining. The idea is that the digital signature changes comparatively slowly during mining and does not result in additional overhead while it still covers important information about the block. For this reason, the modified header could additionally replace the nonce field (and, perhaps, the timestamp) of the block with a fixed value. Similarly to transaction signatures, a block signature could include an additional byte to indicate its type, which would determine the method of calculating the modified block header among several alternatives.

Digital signatures can be effectively used with existing mining pool protocols (e.g., Stratum). In this case, a private key for creating block signatures can and probably should be kept strictly by the pool administrator; pool participants do not need to know the key as they can use digital signatures supplied through the mining pool protocol. The proposed change does not require any changes to the mining software and hardware; indeed, software can be unaware that a part of the coinbase script is allocated to the block signature.

To verify the validity of a block signature, one has to perform the following steps similar to simplified payment verification:

1. verify that the chain of block headers is valid
2. request the coinbase transaction for the block being checked and the respective Merkle branch
3. calculate the modified Merkle root based on the coinbase transaction and its Merkle branch
4. calculate the modified block header based on the modified Merkle root and possibly the type of block signature
5. determine the public key of the block miner based on the coinbase script
6. check the digital signature of the block based on the public key and the modified block header.

The above description can be modified for a case when a miner wants to sign certain transactions in a block instead of all transactions. For example, a miner may want to sign only transactions received from financial institutions via secure channels as described in Section 2.1. In this case, the modified Merkle root value may be built using only these transactions and the coinbase transaction; to enable independent verification of a signature, signed transactions could be placed at the beginning of the block, and the number of these transactions could be included into the coinbase script.

References

[1] *Satoshi Nakamoto* (2008). Bitcoin: A peer-to-peer electronic cash system
URL: `https://bitcoin.org/bitcoin.pdf`

[2] *David Schwartz, Noah Youngs, Arthur Britto* (2014). The Ripple protocol consensus algorithm
URL: `https://ripple.com/files/ripple_consensus_whitepaper.pdf`

[3] Litecoin. In: Litecoin Wiki
URL: `https://litecoin.info/Litecoin`

[4] Ethereum: A next-generation smart contract and decentralized application platform. In: Ethereum project wiki
URL: https://github.com/ethereum/wiki/wiki/White-Paper

[5] *Daniel Larimer, Charles Hoskinson, Stan Larimer* (2014). BitShares: a peer-to-peer polymorphic digital asset exchange
URL: http://scribd.com/doc/173481633/BitShares-White-Paper

[6] Whitepaper: Nxt. In: Nxt Wiki
URL: https://wiki.nxtcrypto.org/wiki/Whitepaper:Nxt

[7] Blockchain.info
URL: https://blockchain.info/

[8] Blocktrail
URL: https://www.blocktrail.com/BTC

[9] *Pieter Wuille* (2012). Hierarchical deterministic wallets (BIP 32)
URL: https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki

[10] *Ilja Gerhardt, Timo Hanke* (2012). Homomorphic payment addresses and the pay-to-contract protocol
URL: http://arxiv.org/pdf/1212.3257v1.pdf

[11] Secret sharing. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Secret_sharing

[12] *Guy Zyskind, Oz Nathan, Alex Pentland* (2015). Enigma: decentralized computation platform with guaranteed privacy
URL: http://enigma.media.mit.edu/enigma_full.pdf

[13] *Dan Bogdanov, Sven Laur, Jan Willemson* (2008). Sharemind: a framework for fast privacy-preserving computations. In: Proc. of 13th European Symposium on Research in Computer Security, ESORICS 2008, LNCS, Vol. 5283, pp. 192–206
URL: http://kodu.ut.ee/~swen/publications/articles/bogdanov-laur-willemson-2008.pdf

[14] *Greg Maxwell* (2015). Confidential transactions
URL: https://github.com/ElementsProject/elementsproject.github.io/blob/master/confidential_values.md

[15] *Robert Sams* (2015). No, Bitcoin is not the future of securities settlement. In: Clearmatics blog
URL: http://www.clearmatics.com/2015/05/no-bitcoin-is-not-the-future-of-securities-settlement/

[16] *Gavin Andresen* (2011). M-of-N standard transactions (BIP 11)
URL: https://github.com/bitcoin/bips/blob/master/bip-0011.mediawiki

[17] *Gavin Andresen* (2014). O(1) block propagation
URL: https://gist.github.com/gavinandresen/e20c3b5a1d4b97f79ac2

[18] *Gavin Andresen* (2013). March 2013 chain fork post-mortem (BIP 50)
URL: https://github.com/bitcoin/bips/blob/master/bip-0050.mediawiki

[19] (2015). Some miners generating invalid blocks
URL: https://bitcoin.org/en/alert/2015-07-04-spv-mining

[20] *Meni Rosenfeld* (2012). Analysis of hashrate-based double-spending
URL: https://bitcoil.co.il/Doublespend.pdf

[21] *Vitalik Buterin* (2014). Proof of stake: How I learned to love weak subjectivity. In: Ethereum Blog
URL: https://blog.ethereum.org/2014/11/25/proof-stake-learned-love-weak-subjectivity/

[22] *V. Gayoso Martínez, L. Hernández Encinas, C. Sánchez Ávila* (2010). A survey of the elliptic curve integrated encryption scheme. In: Journal of Computer Science and Engineering, Vol. 2 (2), pp. 7–13.
URL: http://www.researchgate.net/profile/Carmen_Sanchez_Avila/publication/255970113_A_Survey_of_the_Elliptic_Curve_Integrated_Encryption_Scheme/links/02e7e5212654222f0a000000.pdf

[23] src/rpcrawtransaction.cpp. In: Bitcoin Core Github repository (retrieved on Sep 30, 2015)
URL: https://github.com/bitcoin/bitcoin/blob/master/src/rpcrawtransaction.cpp

[24] src/rpcmining.cpp. In: Bitcoin Core Github repository (retrieved on Sep 30, 2015)
URL: https://github.com/bitcoin/bitcoin/blob/master/src/rpcmining.cpp

[25] Stratum protocol
URL: https://mining.bitcoin.cz/help/#!/manual/stratum-protocol

[26] *Adam Back, Matt Corallo, Luke Dashjr et al.* (2014). Enabling blockchain innovations with pegged sidechains
URL: https://www.blockstream.com/sidechains.pdf

[27] Atomic cross-chain trading. In: Bitcoin Wiki
URL: https://en.bitcoin.it/wiki/Atomic_cross-chain_trading

[28] (2014) Counterparty recreates Ethereum's smart contract platform on Bitcoin. In: Counterparty News
URL: http://counterparty.io/news/counterparty-recreates-ethereums-smart-contract-platform-on-bitcoin/

[29] *Luke Parker* (2015). Rootstock is coming, are Ethereum's days numbered, or will the $18 million dollar idea survive. In: Brave New Coin
URL: http://bravenewcoin.com/news/rootstock-is-coming-are-ethereums-days-numbered-or-will-the-18-million-dollar-idea-survive/

[30] *Joseph Poon, Thaddeus Dryja* (2015). The Bitcoin Lighting Network: scalable off-chain instant payments
URL: http://lightning.network/lightning-network-paper.pdf

[31] *Leslie Lamport, Robert Shostak, Marshall Pease* (1982). The Byzantine generals problem. In: ACM Transactions on Programming Languages and Systems, Vol. 4 (3), pp. 382–401
URL: http://research.microsoft.com/en-us/um/people/lamport/pubs/byz.pdf