Digital Assets on Public Blockchains

White Paper

BitFury Group

Mar 15, 2016 (Version 1.0)

Abstract

Digital asset management is one of promising applications of blockchain technology. Blockchains could provide principal disintermediation between digital asset issuers, application developers and consumers and decouple tasks related to asset management, such as issuance, transaction processing, securing users' funds and establishing users' identities. This paper outlines basic components of blockchain-based asset ledgers, as well as their use cases for financial services and for emerging Internet of Things and consumer-to-consumer markets. We describe existing and prospective deployment models for asset ledgers, including multi-asset blockchains, colored coin and metacoin protocols. This paper focuses primarily on Bitcoin-based services and, to a lesser degree, on public blockchains in general.

Digital assets are a relatively recent development related to the spreading use of information technologies in the financial services. Digital asset is a floating claim of a certain service or goods ensured by the asset issuer, which is not linked to a particular account, and is governed using computer technologies and the Internet, including asset issuance, claim of ownership, and transfer. Digital assets have numerous use cases, including:

- Shares and financial securities
- Smart property
- Tie to a fiat currency
- Local community money
- Coupons
- Digital collectibles
- Access and subscription to certain resources.

Using blockchain infrastructure for digital asset management enables creating pure digital assets (i.e., self-sufficient assets not acting as a "proxy" for real-world assets), which could be considered a transformative technology [1]. Digital asset management could leverage security properties of blockchains, which include:

- Impossibility of counterfeit
- Immutability
- Disintermediation and ease of transfer
- Transparency and ease of auditing
- No overhead related to transaction processing
- Network effect brought by the unified infrastructure for multiple types of tokens

Blockchain-based digital assets (which we call *digital asset coins*, or simply *asset coins*) could be used in both financial contexts and in emerging consumer-to-consumer markets and Internet of Things (IoT).

Blockchain architecture was introduced in [2]. We retain categorization of blockchains from [3]. In particular:

- *Public blockchains* are blockchains that grant read access and ability to create transactions to all blockchain users. Users can transfer value without the expressed consent of blockchain operators. (Note that ordinary end users are not required to store any blockchain data.)
- *Private blockchains* limit read access to the predefined list of entities (e.g., blockchain operators and auditors). End users need to rely on interfaces provided by blockchain operators in order to read and submit transactions.

- *Permissionless blockchains* allow anyone to participate in building the blockchain. The core property of these blockchains is censorship resistance, i.e., any valid transaction broadcast over a permissionless blockchain network would be included into the blockchain. The term "permissionless" corresponds to the fact that there are no blockchain-wide policies restricting the use of the blockchain (although service providers can implement service-specific restrictions). Thus, a permissionless blockchain is by definition free for entry / exit for end users and application developers.
- In *permissioned blockchains*, blockchain building is restricted to a set of known entities. The term "permissioned" reflects the fact that the entities building the blockchain may introduce policies to arbitrarily censor transactions, therefore potentially restricting the blockchain use by end users and application developers.

Public blockchains could be either permissionless (e.g., existing cryptocurrencies) or permissioned (e.g., the federated sidechains concept [4]); private blockchains could only be permissioned. Public permissioned blockchains eliminate possibility of Sybil attacks [5], thus in principle providing a greater degree of scalability and flexibility compared to permissionless blockchain designs. Unlike private blockchains and similarly to permissionless blockchains, the correct operation of blockchain operators in public permissioned blockchains would be mostly mechanical.

The properties of public blockchains – easy entry and exit, openness, transparency, built-in precautions for operation in untrusted environment – could be benefitial for their adoption for decentralized applications. Thus, public blockchains could create ubiquitous infrastructure for the Internet of Value (IoV), with digital assets being one of its core parts. In contrast, private blockchains could retain reliance on trusted third parties for basic operations, thus limiting their innovative potential. For these reasons, our study will be largely focused on public blockchains; for the review of restricted access initiatives, one could refer to the *Permissioned distributed ledgers* report by Tim Swanson [6] (note that the report does not contain information on newer ledger initiatives, such as MultiChain [7] and Openchain [8]).

Among existing permissionless blockchain networks that could be used as a basis for overlay asset protocols, Bitcoin is more secure than alternatives, both in terms of attack costs [9] and intensity of study by cryptographers. While Bitcoin lacks a native support for user-defined assets, digital assets could be implemented with the help of overlay asset protocols – colored coins (e.g., Open Assets protocol) and metacoins (e.g., Counterparty) [10]. These protocols rely on the ability of the Bitcoin protocol to store small amounts of data on the Bitcoin blockchain. An alternative to overlay protocols is blockchains with native support of digital assets, which could be pegged to the Bitcoin ecosystem in terms of currency supply (sidechains) and/or in terms of security (merged mining, blockchain anchoring).

Previous work. Probably the first overview of digital assets (in the form of colored coin protocols) was conducted by M. Rosenfeld in 2012 [11]. Digital assets (more concretely, smart property) tied to the Bitcoin blockchain are researched by A. Mizrahi [12]. The legal aspects of Bitcoin overlay protocols

are the topic of the study by R3 [13].

Our contribution. Our paper mostly systematizes the available information on the use of digital assets with public blockchains. We draw a distinction between institutional and peer-to-peer digital asset use cases, which are sometimes conflated in the literature. We elaborate requirements and main components of public blockchains for use with digital assets and show that public blockchains could be an appropriate technology for emerging consumer-centric, peer-to-peer digital asset markets. We provide arguments for the use of public inclusive blockchain infrastructure for hosting diverse digital assets as opposed to maintaining asset-specific or issuer-specific private ledgers. We also present a voluntary service/customer identification scheme (Appendix A), although it is mostly based on previous research by Bitcoin developers.

Structure of the paper. We discuss basic principles of blockchain infrastructure for digital assets in Section 1 and examine potential use cases for asset coins. In Section 2, we describe the foundations of overlay asset protocols in the Bitcoin ecosystem. We then examine six existing overlay asset protocols together with their applications in Section 3.

1 Digital Assets on Blockchain

First, we must consider the minimum requirements for a ledger for digital assets:

- **User security:** The ledger needs to implement adequate authorization protocols to identify ownership and permit transfer or issuance of assets
- **Counterfeit resistance:** The system needs to have mechanisms to ensure impossibility of counterfeiting assets
- **Auditability:** The system needs to store all asset transactions in order to permit audits of activity (e.g., by regulatory bodies)

There are several additional properties that, while not required, can influence adoption of the system by users and its regulatory compliance, such as internal immutability (i.e., immutability ensured by intrinsic properties of the underlying computing system rather than from the identities of system operators).

One category of digital assets are electronic money [14], where the asset is a claim to a real-world currency. Centralized electronic money systems (PayPal, WebMoney, Google Wallet, Apple Pay, etc.) are commonly used in e-commerce. In essence, a centralized asset transfer system provides its users with a web interface and a database back-end to store account balances and transaction history. The system could use a two-factor password-based authentication vulnerable to various attack vectors such as phishing. The system also provides merchants with proprietary tools (such as APIs/SDKs) in order to accept payments from customers.

Centralized digital asset systems necessitate significant investments into back-end infrastructure, user authentication and regulatory compliance, therefore making them difficult to deploy and maintain for small and medium enterprises (SMEs). The users and auditors of such systems could be concerned about possible mutability of transactions, high availability of the system and its transparency (or lack thereof). These concerns would be higher in the case the system maintainer is a comparatively small company without sufficient public reputation. For similar reasons, centralized digital asset infrastructure (i.e., infrastructure with a single entity controlling all aspects of the asset management) is unlikely to capture consumer-to-consumer markets.

Blockchain-based ledgers provide an alternative to centralized digital asset management. Blockchain technology allows to decouple the basic tasks performed by centralized e-money processors. These tasks include:

- **Transaction processing:** This could be performed in a decentralized manner by geographically distributed nodes of the network. Moreover, defining the rules for transaction processing (i.e., defining valid transactions) could be split from the processing
- **Asset issuance:** In the most general case, this could be performed by any user of the blockchain network.
- **Securing user's funds:** This could be performed by third parties using custodial or non-custodial wallets.
- **Identities of services** (and optionally **customers**): This could be established by building public key infrastructure based on a blockchain
- **Application development:** This does not require cooperation with blockchain maintainers

Thus, blockchains provide a decentralized digital asset management model, which could be less demanding and more appealing for asset issuers, services and customers. For asset issuers, blockchain-based ledgers could be considered a specialized platform as a service (PaaS) [15]. While blockchains are not the only possible type of PaaS for asset management, the core features of blockchain technology, such as increased auditability and user security could make them more attractive than potential general-purpose alternatives. Additionally, the absence of reliance on a single vendor and the associated decreased cost of operations could be a further advantage in the case of permissionless blockchains or blockchains with a diverse participation of transaction processors.

1.1 Constituents

The core constituents of a blockchain-based ledger are as follows.

1.1.1 Blockchain Specification

Specification regulates the way data is transmitted among nodes in the supporting blockchain network and how the state of the blockchain is derived locally based on the received data, i.e., semantics of transactions (these two aspects in blockchains are inseparable by design). The specification includes:

- **Transaction logic:** Valid transactions with regard to the present system state; the rules how transactions transform the system state, etc.
- **Immutability logic:** How transactions are grouped into blocks, and how block headers are secured
- **Consensus logic:** How nodes agree upon the state of the system; how blockchain forks are resolved, etc.
- **Network logic:** How transactions, blocks and other data are transmitted among network nodes, etc.

In principle, rules that are difficult to formalize could be enforced directly by a limited list of asset transaction validators. However, centralized transaction processing would introduce security vulnerabilities. Fully automated transaction processing would be more beneficial in the long run, both from the point of view of security and because of increased compliance, which could be achieved by embedding prerequisites for regulatory compliance (e.g., transaction finality rules and auditability requirements) into the blockchain specification.

In the ideal case, the specification would include all rules of transaction processing, enabling parties to create direct contracts between themselves with the blockchain network specified as a means of value transfer rather than an active party. In this sense, public blockchains could be compared with the Internet as a means of data transfer. The Internet protocols (including application layer protocols, such as HTTP and HTTPS) do not reflect any financial logic. Nevertheless, these protocols are widely utilized in modern electronic financial services facilitating, e.g., end-to-end encryption with the help of HTTPS. Similarly, a public blockchain could facilitate compliance for next-generation financial services without directly implementing service-specific compliance, such as obligatory customer identification, on the blockchain-wide level[1].

A publicly available blockchain specification together with open access to blockchain development tools could create an optimal environment for innovations and third-party applications. While managing a public specification could be more difficult than a proprietary specification (e.g., from the point of view of backward compatibility), public specification aligns with the overall spirit of blockchains as consensus-driven systems.

1.1.2 Blockchain Notaries

An asset issuer using blockchain infrastructure is not generally required to process transactions or to write data to the blockchain – this task could be delegated to *blockchain notaries*. Notaries could be either known entities (in permissioned blockchains), or any users satisfying technical capabilities

[1] A requirement to have specific built-in regulation-related policies in the case of public blockchains could be counterintuitive and quite similar to a theoretical requirement of having anti-piracy policies embedded into the Internet protocol stack. In both cases, proposed policies **(i)** are outside of the scope of the protocol; **(ii)** are jurisdiction-specific, while the protocol is inherently global; **(iii)** contradict the separation of concerns principle [16].

imposed by a blockchain consensus algorithm (in permissionless blockchains). Permissioned blockchains could be more beneficial for financial institutions in the short term because of the flexibility of the blockchain specification and increased compliance. On the other hand, permissionless blockchains could prove more attractive for consumer-to-consumer markets and IoT applications because of inherent trustlessness and permissionless entry and exit.

Permissionless blockchains (such as cryptocurrencies) necessitate rewards for users participating in building and securing the blockchain. This goal is accomplished by introducing tokens, which are generated by creating blocks, and/or by collecting transaction fees. In the case of a permissioned blockchain, blockchain notaries are interested in keeping the blockchain safe, as it provides them with a stream of revenue, e.g., by running services in top of it.

1.1.3 Blockchain Network

A public blockchain network provides three security modes for constituent nodes:

- **Full verification** nodes that verify and store every transaction circulating in the network. This security mode could be used by blockchain notaries, regulators, auditors, analytical services and dedicated "blockchain as a service" providers
- **Simplified payment verification** (SPV) nodes [2], which would be used by a vast majority of end users, as this security mode requires little computational resources and memory space
- **Partial verification** nodes made possible with the help of segregated witness and fraud proofs [17]. These nodes could verify a small percentage of transactions (e.g., 1%), while contributing to the overall security of the blockchain network. Partial verification nodes could be operated by service providers on the blockchain

In the case of a blockchain with restricted read access, the architecture of the blockchain network would be determined by transaction processors. For example, transaction processors could operate full nodes, and all other users could be provided to concerning transactions either through SPV network nodes or through equivalent web application interfaces. Thus, blockchains with restricted access could be less scalable or reliable because of uneven distribution of transaction processing.

There is an important distinction between SPV nodes and web API access to blockchain data. While SPV nodes do not increase the security of the blockchain network, their use together with the publicly available chain of block headers could provide the following properties:

- **Uniqueness:** There would be a single copy of a blockchain; multiple copies could be trivially detected and could only be attributed to the active collusion of blockchain notaries
- **Immutability:** The blockchain could not be retroactively changed by the collusion of notaries, as such a change would not be accepted by SPV nodes.

Note that these properties could alternatively be achieved by making block creation expensive, e.g., with the help of proof of work.

In the case that access to the blockchain is provided via web APIs without disclosing the blockchain structure, reliably proving uniqueness and immutability becomes more difficult. Even if the regulator or an auditor would have complete access to the blockchain (e.g., by operating a full verification node), data provided to the regulator could differ from data served via API as a result of an eclipse attack [18] performed by colluding blockchain notaries.

1.1.4 User Authentication and Authorization

User authorization in blockchains is performed using public key cryptography. In the simplest case, blockchain-based assets are bearer assets; i.e., the ownership of an asset is determined by the knowledge of a private key. Two-factor authentication or other security measures comparable to those of centralized e-money systems [19] could be implemented by using dedicated wallet services. A Bitcoin-like blockchain scripting language could allow both custodial and non-custodial wallets (e.g., implemented with the help of 2-of-3 multisignature scheme [20]). Security properties of public key cryptography could be boosted by the use of specialized *hardware wallets* for signing transactions. Overall, blockchain infrastructure provides security decentralization and eliminates single points of failure inherent to centralized e-money ledgers.

In order to maintain user privacy, blockchain users could utilize hierarchical deterministic wallets [21] and the pay-to-contract protocol [22], which allow for the creation of publicly unlinkable addresses supporting on-demand auditing. Transaction amounts could be masked using range proofs [23]. In the case of more complex transaction models, e.g. for smart contracts, zero-knowledge proofs [24] and secure multi-party computations [25] could be used in order to execute contracts while not disclosing data to any of computers (see, e.g., Enigma project [26] and Zerocash [27]).

As blockchain infrastructure provides a complete time ordering of events, it could be used to implement decentralized public key infrastructure (PKI), which would link identities of persons and entities to their public keys. Public key infrastructure could be organized as a part of the blockchain specification, or as a separate overlay protocol (similar to colored coin protocols). PKI would allow for legally recognized value transfer and asset issuance[2]. See Appendix A for a high-level outline of a possible blockchain PKI implementation.

1.1.5 Asset Issuance

In general, assets could be issued by any blockchain user; the semantics of assets would be imposed by the issuer. As asset issuance is a special type of transactions, the identity of the issuer could be determined according to the general user identification rules (using the blockchain-based PKI or other techniques [29]). A regulatory body could explicitly acknowledge asset issuance by co-signing the corresponding transaction together with the issuer, or by granting the issuer a special kind of the digital certificate.

[2] Large-scale legally binding public key cryptography systems already exist, e.g., ID cards in Estonia [28] and other European countries.

Asset issuance could specify, besides the type and amount of issued assets and the identity of the issuer, other asset properties:

- An asset could be marked as *locked*, meaning the assets of the same type cannot be issued in the future by anyone, including the initial issuer. This type of assets is useful, e.g., for creating non-dilutable shares
- An asset could be marked as *divisible* to several decimal places (cf. with Bitcoin, which is divisible to 8 decimal places)
- An asset could be made *non-transferable* in order to limit secondary market (e.g., due to regulation requirements)
- Additional metadata could be associated with the asset, either directly or in the form of a hash commitment. In the second case, off-chain data could be retrieved with the help of distributed hash tables, e.g., implemented using BitTorrent protocol. Metadata could be useful, e.g., in implementing event tickets

1.2 Deployment Models

1.2.1 Separate Blockchains for Assets

Each digital asset or a set of assets maintained by the same issuer could potentially have its own blockchain, either permissionless or permissioned. Securing a small-scale permissionless blockchain could prove expensive, as the cost of an attack on the system is proportional to the cost of the blockchain token. (Merged mining [30] eliminates most technical hurdles with security, as it allows securing multiple blockchains with the same computational resources. On the other hand, merged mining in a permissionless environment could be unsafe, as an attacker with enough hash rate could deliberately mine empty blocks or otherwise disrupt transaction processing.) A permissioned blockchain could be more resilient to attacks, but it would still have a single point of failure in the form of a single transaction processor.

From the auditing and regulating points of view, properties of an issuer-managed blockchain could be similar to existing asset management systems. Because of the centralization, the asset issuer acting as a blockchain operator may have an incentive reporting unauthentic information during audits.

In the case there are established asset trade pairs, using separate blockchains for each asset could be inefficient. The cost of operating an issuer-specific blockchain (either on-site or using a PaaS) could be comparable to traditional asset management systems because of the need to develop end user applications (such as wallet services with secure authentication), accounting tools, etc. Additionally, using separate blockchains could complicate the development of third-party applications and diminish the network effect by requiring additional tools to interact with other digital assets.

1.2.2 Colored Coin Protocols

A colored coin protocol is an overlay protocol on top of a blockchain that does not support user-defined digital assets natively. Colored coin protocols share the user authentication model with the underlying blockchain (see Section 2). However, because the validity of colored coin transactions is not checked by the blockchain network, colored coin protocols lack efficient payment verification methods (cf. with simplified payment verification in Bitcoin).

Colored coin protocols using the Bitcoin blockchain include ChromaWay [31], Open Assets [32] and Colored Coins Protocol [33].

1.2.3 Metacoins

A metacoin system is a colored coin protocol coupled with a middleware layer in the form of dedicated servers, which verify colored coin transactions. A metacoin system could provide automated order matching for trading asset pairs, dividend payments, and so on. Metacoin systems may utilize a dedicated cryptocurrency as a means of payment for provided services.

Metacoin systems on top of the Bitcoin blockchain include OmniLayer [34], Counterparty [35] and CoinSpark [36].

1.2.4 Multi-asset Blockchains

Multiple assets can be natively supported by a blockchain. Compared to other deployment models, multi-asset blockchains have more space-efficient proofs of ownership, as simplified payment verification [2] could be utilized for all natively supported blockchain assets. On the other hand, known mechanisms of sharing blockchain security (merged mining and blockchain anchoring) pose security risks in permissionless context.

Blockchains with the federated governance model could eliminate aforementioned security risks. For example, merge-mined blocks could be digitally signed by the miner (in the case a contract is established between Bitcoin miners and blockchain maintainers) or by the blockchain maintainers themselves. Similarly, anchor transactions could be produced by known parties. Alternatively, a blockchain with shared security could implement an alert system, which would notify users of blockchain attacks and halt operations accordingly. The federated governance model puts the greater responsibility on the blockchain maintainers. As the maintainers can effectively determine the state of the blockchain, they could be legally obliged to be able to reverse transactions, freeze funds, etc. by the regulatory bodies.

A multi-asset blockchain could be integrated into existing blockchain infrastructure by using sidechain technology [4]; Elements Alpha [37] developed by Blockstream is an example of Bitcoin-pegged multi-asset blockchain. Independent multi-asset blockchains include Nxt [38] and BitShares [39].

1.2.5 Smart Contract Blockchains

User-defined assets could be represented with the help of a smart contract on a smart contract blockchain. The contract could store the mapping of the addresses of current holders of the asset to the corresponding balances [40]. These balances could be updated with the help of messages sent to the contract encoding asset transfer or issuance. The contract could use the conventional authorization scheme of the underlying blockchain in order to check transfer and issuance permissions, or could specify new rules for asset transactions.

Ethereum [41] is an example of an independent stat smart contract blockchain. Rootstock [42] is a conceptual smart contract blockchain pegged to Bitcoin.

1.3 Use Cases

Digital asset blockchains could be utilized by a variety of users and applications. We single out the following categories of blockchain users:

- asset issuers
- blockchain notaries
- regulators
- user application developers
- end customers.

In certain cases, a user might belong to several categories (e.g., in the case of an in-application digital asset, application developers would simultaneously be asset issuers). However, one of the advantages of blockchain technology is that the specified user roles could be clearly separated; e.g., an asset issuer could delegate transaction processing and application development to third parties. Moreover, the public blockchain environment could provide capabilities such as the secondary asset market and third-party application development, without any actions required from the asset issuer.

Naturally, different categories of user would have differing requirements as to the operation of a blockchain (Table 1). The requirements would also depend on the nature of digital assets recorded on the blockchain. For example, legality concerns for digital securities would be higher than for other assets, and the entry barrier for these types of digital assets is expected to be quite high.

In general, digital asset use cases fall into one of two categories:

- **Institutional assets,** which are characterized by institutionalized transaction processors and the legal requirements taking precedence of ease of entry and global reach. Digital assets that represent securities would generally fall into this type.
- **Peer-to-peer assets,** with the underdeveloped or non-existent market of dedicated transaction processors and a strong requirement of easy entry and global reach of technology. This type of digital assets would include in-application assets, business-to-consumer assets (e.g., discounts, gift cards), content subscription assets, etc.

Table 1: Generic requirements and concerns for different users of blockchain technology. Requirements are given in no particular order

User category	Requirements
Asset issuers	Counterfeit resistance, entry permissions, cost of operation, openness (= network effects and third-party applications)
Blockchain notaries	Cost of operation, entry permissions, definitiveness and completeness of transaction processing rules[3]
Regulators	Auditability, transaction finality, immutability
Application developers	Ease of application development (including availability of manuals, APIs, SDKs, and frameworks, roadmap of technology, etc.), entry permissions, reach
End users	Ease of use, user security, entry permissions, confidentiality, transparency, reach, legality

In the case of smart property, the categorization is unclear. There are institutional registries for certain types of property (e.g., real estate); however, for most property, centralized ownership registries do not and, arguably, should not exist.

Regulatory requirements for institutional assets could necessitate the use of private or strictly regulated public permissioned blockchains, which would be maintained by existing transaction processors. In this case, blockchain technology could provide an innovative application deployment model (distributed database and code base shared among participants), built-in audit trails and, possibly, more third-party participation (e.g., in the form of independent authentication services). In contrast, peer-to-peer assets could productively use public blockchains because they cover the requirements of easy entry and global reach, while the cost of operation would be low for asset issuers and application developers.

1.3.1 Complex Financial Assets

Digital assets could represent publicly traded financial assets (e.g., securities). These assets require a high level of security, are heavily regulated and used in business-to-business contexts, therefore requiring permissioned blockchains, at least in the short term. Additionally, many kinds of securities could benefit from extensive smart contract capabilities, which are generally not used by other types of digital assets. Financial assets could be traded in a decentralized manner without requiring intermediaries (although wallet services described above could provide additional security.)

Permissionless blockchains could be useful for novel financial services, such as crowdfunding. For example, a company could issue digital assets representing its shares and sell these shares in a crowdfunding or a venture campaign. Later, the company could pay proportional dividends to the holders of its shares.

[3] There should be a clear set of transaction processing rules, ideally formalized in computer code. A blockchain notary should not be held liable for adhering to these and only these rules. Cf. with liability of Internet service providers for transmitting data related to illegal activities.

1.3.2 Smart Property

Smart property represents the ownership of real-world objects with the help of blockchain data. For example, a blockchain-enabled car would operate only if the driver holds the blockchain-based ownership token. The owner could use an application on his smartphone to connect to the car via NFC. The ownership defined in this way could be transferred using a transaction with an input bearing the token.

Smart property assets would have slow transaction velocity and would require security before scalability. Therefore, smart property could plausibly be implemented with the help of dedicated ownership protocols (not yet developed) on top of Bitcoin or other highly secure public blockchains, which do not necessarily support the concept of smart property natively.

Protocols similar to smart property could be implemented for other use cases, e.g., for verifying authenticity of goods [43, 44].

1.3.3 Electronic Money

Digital assets could represent e-money, such as alternative currencies (e.g., local currencies or in-game currencies) or claims of fiat money. Electronic money pegged to real-world currencies generally have high transaction velocity; therefore, they would require scalable, high-throughput infrastructure provided by multi-asset blockchains. Currencies with lower transaction velocity (e.g., local currencies) could use multi-asset blockchains, colored coin protocols or metacoins.

1.3.4 Business-to-Consumer Assets

Digital assets could be used to represent discount, coupons, vouchers, gift cards, etc. The assets would be issued by a merchant and transferred to buyers during purchases; the merchant would define a transparent set of rules of how assets can be redeemed for goods. A large retailer could issue multiple types of tokens and track their distribution and ownership, which would be useful for analyzing the customer base. Compared to existing implementations, blockchain infrastructure would provide a built-in secondary market for assets (although asset transfer could be restricted with the help of issuance metadata).

1.3.5 Event Tickets

A cinema, theater, or concert hall could issue digital assets that correspond to tickets for a specific event. This would allow customers to buy or sell their tickets securely and fast, not being afraid of counterfeit. To prove the ownership of a ticket, a person attending the event would send it to the designated address; this logic could easily be implemented as a mobile application. By adding metadata to coins, the issuer could encode information about a specific ticket, such as a theater seat.

1.3.6 Digital Subscription

Digital assets could be used to monetize access to digital resources, such as stream content. For example, an Internet radio could provide monthly or yearly unlimited access to music expressed as a digital token, which customers can buy for a certain amount of money. Because of the transparency of blockchains, the content provider could easily check when the user's token was issued and whether it is still valid. The provider could issue multiple types of tokens that correspond to various levels of access (read/write, or read-only), or to the access to specific resources or types of resources.

Similar to digital subscription, non-transferable digital assets could be useful for role-based authentication (e.g., a specific type of tokens can be used to identify administrators of a web service.)

1.3.7 Digital Democracy

Digital asset coins can be used to implement voting by sending tokens to the one of several designated addresses. While the existing digital asset systems are not secure enough to hold government elections, they can be used for voting among shareholders or in contests; in the latter case, voting process is easily monetized. Governmental voting would arguably require more complex techniques such as Paillier cryptosystem [45] used together with zero-knowledge proofs in order to enforce anonymity and non-malleability.

2 Overlay Asset Protocols in Bitcoin

The degree of adoption of permissioned blockchains in business-to-consumer and especially consumer-to-consumer applications depends on the entry cost and regulatory pressure. Due to compliance inertia, we expect permissionless or loosely regulated permissioned blockchains to play a significant role in emerging IoT and consumer-to-consumer markets.

Currently, permissionless multi-asset blockchains are either at early stages of development (Elements Alpha by Blockstream) or have significantly lower reach than the Bitcoin ecosystem (Nxt, BitShares). Similarly, smart contract blockchains such as Ethereum and Rootstock are not currently ready for the use in production. Consequently, Bitcoin overlay asset protocols (Table 2) are currently a primary means of managing digital asset coins. Multi-asset blockchains and smart contract blockchains (optionally pegged to the Bitcoin blockchain in terms of anchoring and/or currency supply) could become a viable alternative to these protocols in the future.

2.1 High-Level Architecture

The state of the Bitcoin system at any moment is the collection of unspent transaction outputs (UTXO). Each of transaction outputs has two main parts:

- A non-negative value associated with the output. In Bitcoin, it is an unsigned 8-byte integer equal to the number of satoshis (10^{-8} bitcoins) the output holds.

Table 2: Overlay asset protocols on the Bitcoin blockchain

Name	Website	Year of foundation	Protocol type
ChromaWay	chromaway.com	2012	colored coin
Open Assets Protocol	github.com/OpenAssets	2013	colored coin
OmniLayer	omnilayer.org	2013	metacoin
Counterparty	counterparty.io	2014	metacoin
CoinSpark	coinspark.org	2014	metacoin[4]
Colored Coins Protocol	coloredcoins.org	2015	colored coin

- A locking script used to determine the person(s) eligible for spending the output. The script written in the Bitcoin scripting language [46], which describes the execution of a program on a stack machine [47]. The capabilities of the scripting language are severely limited compared to general purpose programming languages; for example, it lacks loops, I/O, and persistent states.

Instead of explicitly storing system states, the blockchain stores atomic changes between system states – *transactions* [48]. This architecture allows for the restoration of any intermediate state since the inception of the system and up until the current state. Each transaction consists of zero or more transaction inputs, and one or more transaction outputs. A transaction input references a previously unspent output of one of the transactions recorded on the blockchain[5]. An input also needs to supply an unlocking script corresponding to the locking script of the referenced output, which proves that a person spending the output is authorized to do it. A locking script and the corresponding unlocking script are jointly executed on a stack machine; the status of verification depends on the end state of the machine. Locking scripts commonly reference public keys on the secp256k1 elliptic curve [49]. Unlocking scripts provide digital signatures of transactions corresponding to these keys.

The idea behind colored coins and metacoins is to use the Bitcoin blockchain for multiple types of assets and at the same time share their cryptographic security and immutability of transactions. Thus, secondary assets can be implemented without the need to create separate blockchains, which is beneficial from the point of view of security (assuming the underlying blockchain is secure) and development efforts. In addition, digital assets can leverage advanced tools of the underlying system, such as multisignatures in the case of Bitcoin.

An overlay asset protocol provides a generic encoding, which allows associating certain Bitcoin transaction outputs with non-zero amounts of user-defined assets. The distribution of amounts and types of assets associated with transaction outputs is inferred based on additional data encoded into a transaction, and on assets associated with referenced UTXOs. Thus, the encoding essentially creates a virtual blockchain for each user-defined asset with locking and unlocking scripts shared with the Bitcoin blockchain. In contrast, transactions in multi-asset blockchains explicitly specify the asset type

[4] Does not have a native currency

[5] We do not consider specialized inputs such as the first input of coinbase transactions in Bitcoin. These inputs do not reference unspent outputs and do not use the common protocol to prove ownership.

associated with each transaction output. The security of an overlay protocol is strictly equal to that of the underlying blockchains, while multi-asset blockchains need security of their own.

The rules on the asset transfer could be similar to those of Bitcoin, i.e., the sum of values of transaction outputs must not exceed the sum of values of transaction inputs. As for asset issuance, it is commonly restricted by determining the asset type based on the properties of the issuance transaction:

- **for unlocked assets:** based on the public key of the issuer (usually implemented as the public key referenced in the first input of the transaction)
- **for locked assets:** based on the transaction hash

The implicit nature of asset values in overlay asset protocols means that it is generally impossible to determine values of user-defined assets associated with unspent outputs based on the current state of the blockchain system alone; the entire transaction history resulting in the current state is required. Metacoin protocols deal with this issue by introducing a middleware layer storing secondary asset values for unspent outputs and updating them according to new transactions on the blockchain.

The Bitcoin protocol is unaware of movement of any asset other than bitcoins. Thus, nothing prevents a node from broadcasting a valid Bitcoin transaction, which contains an invalid encoding of the asset issuance or transfer, or adding such a transaction into the blockchain. Most overlay asset protocols postulate that a transaction invalid from the point of view of a user-defined asset destroys all value associated with its inputs.

2.2 Encoding Asset Transactions

There are several methods for storing data in Bitcoin transactions [50]:

- **Output value** and **input sequence numbers** allow for storage up to 4 bytes
- **Address field** (20 bytes) can be faked or brute-forced to store the required data
- 1-of-n **multisignature scheme** [51] can be used with one real address and the rest of the addresses containing encoded data. The scheme allows to store $32(n-1)$ bytes of data
- **Return instruction** followed by data can be used in a locking script

The term “colored coins” comes from the older implementations where an asset was tied to the value of transaction outputs. Most modern methods use **RETURN** instruction of the Bitcoin scripting language or multisignature addresses instead. The **RETURN** instruction terminates the verification of ownership immediately with the failure status. Thus, an output with a script containing this instruction is provably unspendable: there exists no unlocking script satisfying the verification process. In the case of digital asset coins, a **RETURN** instruction is followed by an instruction to store certain data in the stack (note that this instruction is never executed during the verification process).

To be considered standard, a transaction needs to have no more than one output with the **RETURN** instruction. Non-standard transactions are commonly not relayed by Bitcoin nodes; most Bitcoin notaries currently create blocks with standard transactions only. Additionally, in order for a transaction to be considered standard, the length of the data used with the **RETURN** instruction cannot exceed certain length. The default length, which is usually used by Bitcoin transaction notaries, was increased from 40 to 80 bytes in Bitcoin Core 0.11 released in July 2015 [52]. Older overlay protocols were developed with the limit of 40 bytes in mind, which is why the value of 40 bytes will be used in the following statement.

An asset transaction using a **RETURN** instruction typically consists of one or more ordinary inputs that unlock unspent outputs tied to a certain Bitcoin address (Fig. 1). Transaction outputs, except for the output with a **RETURN** operation, are ordinary Bitcoin outputs that lock funds on Bitcoin addresses. An asset transaction may move value in bitcoins, just like an ordinary transaction. The main difference between an ordinary Bitcoin transaction and an asset transaction is the **RETURN** output that encodes how digital asset coins are moved and/or created.

Inputs	Outputs
Input0 (0 BTC) **Address:** Alice **Assets:** 10 Asset A	Output0 (0 BTC) **Address:** Bob **Assets:** 10 Asset A
	Output1 (0 BTC) **Script:** **RETURN** ⟨ID⟩ (move 10 Asset A from Input0 to Output0).

Figure 1: The generic form of an asset transaction using a **RETURN** instruction. For the sake of simplicity, we ignore Bitcoin transaction fees in this and the following examples unless specified otherwise; they can be accounted for in one of the existing inputs or in a new transaction input

The data after the **RETURN** instruction usually starts with a short byte sequence enabling the network to identify asset transactions, which are following the overlay asset protocol. The data specifies one or more issuance or transfer instructions. Each of the instructions contains the kind and the amount of assets moved between a certain input of a transaction and a certain output; e.g., an instruction can specify to move 10 units of Asset A from the first input of the transaction to the first output. The digital asset system can check that inputs of the transaction contain necessary amount of asset coins by examining the asset transaction history. Transaction outputs that hold asset coins usually don't have any Bitcoin value or have just enough value to pay transaction fees: there is no reason to couple values of multiple asset types in a single output if they can be spent separately.

Rules of overlay asset protocols are enforced neither by ordinary Bitcoin nodes nor by miners. In this sense, asset coins are invisible to the Bitcoin protocol; a user could spend a UTXO holding asset coins in a transaction considered incorrect from the point of view of the overlay asset protocol. In most overlay asset protocols, all digital asset coins connected to the spent UTXO are considered permanently

lost in this case. The risk of mishandling asset coins is diminished by using specialized asset protocol-aware wallets and other software tools.

2.3 Extensions

2.3.1 Payment Channels

Peer-to-peer payment channels such as Lightning [53] could be used either on top of overlay asset protocols or on top of multi-asset chains to provide a framework for instant transfers and micropayments in digital assets. (Note that Lightning currently does not currently have an efficient way to introduce an asset into the open payment channel, so it cannot be used for instant exchange.) There are other possible designs of the instant transactions layer, such as a two-phase commit protocol [54] to be used in Liquid, the first application of sidechain technology [55]. The important advantage of Lightning Network design is its trustlessness, which assists in building peer-to-peer networks among end users with inherent resilience to misbehaving and non-responding nodes.

2.3.2 Unspent Output Commitments

In order to boost scalability, an overlay asset protocol could be augmented with periodic commitments of unspent outputs bearing a non-zero amount of asset coins [56]. The architecture of these commitments would be similar to proposed Bitcoin UTXO commitments [57]. Using UTXO commitments simplifies verification of asset transactions: it suffices to trace the history of assets since the latest commitment, not since their issuance transaction(s). Commitments could be performed by the asset issuer for specific types of digital assets, or by a trusted third party for all assets using the overlay asset protocol in question. Witness data corresponding to commitments (UTXO properties and a Merkle branch linking them to the commitment) could be served using distributed hash tables.

3 Existing Protocols and Applications

3.1 ChromaWay

ChromaWay is an early colored coin platform developed since 2012. ChromaWay uses the value of transaction outputs and input sequence indices to encode information about asset coin transactions. There are several methods for encoding information (called colored kernels within the platform), including the enhanced padded order-based coloring protocol (EPOBC) [58] and ITOG [59]. Use of transaction output values limits the capabilities of the platform for handling large amounts of asset coins compared to other protocols described in Section 3.

ChromaWay defines two kinds of digital asset transactions that can be identified by a sequence number of the first transaction input:

- **Genesis transactions** create new asset coins. The number of coins created is determined by the value of the first output of the transaction

- **Transfer transactions** move digital assets from inputs to outputs of the transaction. Values of the transaction outputs are used to determine type and amount of asset coins associated with each output

As there are lower limits on values of transaction outputs (546 satoshi at the time of writing) [60], ChromaWay encoding protocols use *padding* – a fixed number added to each output value. In EPOBC, padding is determined based on the sequence number of the first transaction input.

Each transaction output in ChromaWay can be associated with a single type of asset tokens. All assets in ChromaWay platform are locked, i.e., it is impossible to issue more assets of the same type in the future. The reason is that the type of asset tokens is determined by the corresponding genesis transaction.

3.2 Mastercoin / Omni

Mastercoin was developed in late 2013 as one of the first attempts to provide a higher layer protocol on top of the Bitcoin blockchain. In March 2015, Mastercoin was rebranded as OmniLayer.

OmniLayer stores data in Bitcoin transaction using fake addresses or 1-of-3 multisig addresses. The layer supports a number of transaction types, which can be used to create and distribute user-defined assets. Issued tokens can be either divisible to eight decimal places (like Bitcoin), or indivisible. Assets can be issued with one of two instructions:

- Issue a fixed amount of tokens
- Issue tokens in the process of crowdsale

In the second case, the number of issued tokens is proportional to the received funds during the crowdsale. The issuer can claim a certain percentage of issued coins for itself.

All user-defined currencies together with Mastercoin can be sent from one address to another with no fees other than Bitcoin transaction fees. Additionally, OmniLayer provides a decentralized exchange service which allows for the placing of buy or sell orders for any user-defined currency, Mastercoin, and Bitcoin. Orders matched by the service are executed automatically. There exists a special instruction to distribute a certain amount of user-defined currency proportionally among all holders of the currency; this is useful for paying dividends.

3.3 Open Assets

Open Assets Protocol (OAP) was introduced in 2013 and is currently supported by the asset coin wallet Coinprism.

To conform to the OAP specification, a Bitcoin transaction needs to have a special output called the marker output that contains data specifying the redistribution of assets. Data is embedded into the marker output with a **RETURN** instruction. The marker output is used to recognize OAP transactions among ordinary Bitcoin transactions. All outputs of the OAP transaction excluding the marker output either issue or redistribute assets. Each of these outputs has two associated characteristics:

- **Asset ID,** the 20-byte identifier of an asset similar to a Bitcoin addresses
- **Asset quantity** determining the amount of the asset stored in the output

The type of an output is determined by its position in the output list relative to the marker output:

- Outputs before the marker output issue new assets. The identifier of the issued assets is determined by the first input of the transaction.
- Outputs after the marker output transfer assets. The asset ID is determined based on transaction inputs using a complex method (*order-based coloring*).

The quantity of assets does not depend on the built-in Bitcoin value of the outputs. This allows mixing transfer of bitcoins with asset transfer.

3.4 Counterparty

The Counterparty platform offers a variety of services built on top of the Bitcoin blockchain; it uses the blockchain for the reliable publication and timestamping of its messages. Counterparty services are monetized with a native currency, XCP. Like in Bitcoin, the supply of XCP is limited; however, all supply of XCP is pre-mined and was created using the "burning" process [61]. Counterparty stores data in Bitcoin transactions using several methods, depending on the size of data:

- 1-of-3 multisignature addresses where the first address is the real address of the sender, and the other two encode data
- Data after a **RETURN** instruction
- Fake addresses

All data is secured using encryption and is identified with a predefined prefix.

One of the core services of the platform is the management of digital assets. The Counterparty protocol allows users to manage certain properties of assets, such as divisibility and callability. A divisible asset is divisible up to 8 decimal places, like Bitcoin. A callable asset can be repurchased by the issuer from its owners at a fixed price on a specified date. A newly created asset can be made locked against the further issuances.

Counterparty supports several basic instructions (messages) to govern the flow of assets including, if not stated otherwise, XCP and bitcoins:

- **Send** message sends the specified quantity of any Counterparty asset from the source address to the destination address
- **Order** message allows placing public buy / sell orders over the blockchain. The Counterparty platform automatically matches orders and completes the trade if the order doesn't involve Bitcoin
- **Cancel** message is used to revoke open orders

- **Issue** message is used to issue user-defined digital assets, create new digital assets, or change metadata associated with an asset
- **Dividend** message distributes some quantity of a Counterparty asset among holders of a certain user-defined asset proportionally to their holdings

The possibilities of the platform would be further boosted by incorporating Ethereum smart contracts engine [62] (as of the time of writing, the smart contracts engine is available for testing only [63]). The engine would allow manipulating Counterparty assets algorithmically.

3.5 CoinSpark

CoinSpark is an overlay asset protocol developed by Coin Sciences. Like OmniLayer and Counterparty, CoinSpark relies on a middleware layer in the form of tracking servers; however, the system does not feature the internal currency. CoinSpark provides three geographically redundant tracking servers, as well as source code for building custom servers.

The protocol discerns between two types of transactions: genesis transactions creating new assets, and transfer transactions. Data about asset issuance or transfer is encoded with the help of the **RETURN** instruction. CoinSpark uses modified Bitcoin addresses for transactions to ensure that each receiving address of an asset transaction is aware of the transaction semantics.

Every CoinSpark asset needs to be backed by a web page, URL of which is specified during its genesis transaction. A web page provides detailed information about the asset, such as its issuer, description, issue date, associated icon, etc. in JSON format. Some of the fields can be changed; others such as asset name are fixed by calculating a hash of data structure corresponding to the fixed fields and comparing it to the asset hash which was obtained during asset creation. One of more interesting parameters of CoinSpark assets is automatically charged transaction fees.

Unlike other overlay asset protocols that use tags to identify asset transactions, CoinSpark has a less strict method of mapping Bitcoin transactions to asset transactions. By default, any CoinSpark assets associated with the inputs of a transactions are moved to the last transaction output not locked with a **RETURN** instruction [64].

3.6 Colored Coins Protocol

Colored Coins Protocol (CCP) is maintained by Israeli startup Colu and has been fully open-sourced since June 2015.

Compared to other protocols, CCP offers several additional capabilities:

- Colored coin data can be stored with either the **RETURN** instruction, or 1-of-2 or 1-of-3 multisignature addresses. Furthermore, additional metadata can be stored utilizing the BitTorrent protocol
- Metadata can be attached to any asset transaction to control the peculiarities of issuance and transfer operations (e.g., a list of possible recipients)

Colored Coins Protocol is at public beta stage; significant design modifications could be expected in the future.

Compared to the other implementations, the main advantage of Colored Coins Protocol is metadata that can be attached to any transaction. Metadata consists of two parts:

- **Static data** contains information about the issued asset, similar to Counterparty assets, and free-form user data, which can be used to couple digital tokens with arbitrary data (e.g., seat number for an asset representing a theater ticket)
- **Rules** describe restrictions on asset transfer and/or issuance, enabling asset-specific smart contracts. For example, a rule can specify an expiration date of the asset, or a list of addresses it can be transferred to, or fees that need to be paid for a transfer. Rules are inherited upon asset transfer

Additional data widens possible domains of colored coin applications:

- Rules can specify a small fee payment to the designated address each time the token changes hands. This is useful for coins representing event tickets and other cases when the seller could be concerned about the secondary market of coins. The inheritance settings of the rule can prevent a token owner from changing metadata associated with the token.
- CCP allows to specify explicitly a list of addresses of possible shareholders, which is useful for creating KYC-compliant digital assets. The inheritance settings of the holder rule can be set to allow trustful shareholders to loosen the restrictions by including more addresses in the list or removing the restriction rule altogether. Thus, rules act similarly to smart contracts; however, this approach is not as flexible as native smart contract capabilities provided, e.g., by Ethereum.

3.7 Asset Coin Applications

Table 3: Some asset coin applications using Bitcoin infrastructure

Name	Category	URL	Protocol
Linq	equity market	n/a	Open Assets[6]
MaidSafeCoin	cloud platform	maidsafe.net	OmniLayer
Tether	money transfer	tether.to	OmniLayer
Get Gems	social network	getgems.org	Counterparty
Storj	cloud platform	storj.io	Counterparty
Cuber	money transfer	cuber.ee	ChromaWay
Koinify	crowdfunding	koinify.com	Counterparty
Swarm	crowdfunding	swarm.fund	Counterparty

[6] Uses a private fork of the Bitcoin blockchain

3.7.1 Linq

Linq [65] is the trading platform developed by NASDAQ for their Private Markets application. The main goal of Linq is to simplify record-keeping and to improve auditability with the help of blockchain technology. The platform currently has a limited trial run; in the future, it would provide experience mirroring that of traditional stock exchanges.

Unlike applications described below, Linq does not utilize the Bitcoin blockchain directly, but rather relies on a privately operated blockchain fork.

3.7.2 MaidSafeCoin

MaidSafeCoin is an intermediary currency, which was used to fund MaidSafe. MaidSafe is an open source decentralized Internet platform. Instead of specialized servers, the MaidSafe network uses computers of its users for data storage and processing. Client applications can use the network for cloud storage, encrypted messaging, hosting web sites or distributed databases, processing documents or any other data, trading, etc. To incentivize end users and developers, MaidSafe uses the internal currency, Safecoins:

- Users who provide their resources for the needs of network are rewarded in safecoin
- Developers earn safecoins in proportion to how often their applications are used

Safecoins are managed by the MaidSafe network; the currency itself is not tied to the Bitcoin blockchain in any way, so it cannot be called a digital asset coin. However, 10% of safecoins were distributed in a form of the intermediary currency, MaidSafeCoin, with the help of Mastercoin-based crowdsale.

3.7.3 Tether

Tether USD (formerly Realcoin) is an OmniLayer-based digital asset issued by Tether. Tether USDs are backed 1-to-1 by US dollars held in the company's reserves; the rate of the currency is held constant at $1 USD. Tether tokens are used by the company to provide global instant transfers.

3.7.4 Get Gems

Get Gems is a social messaging application for iOS and Android. The internal currency, Gems, is used to monetize provided services: while sending content between friends is free, sending unsolicited messages or advertisements costs Gems. Gems is a locked currency with 100 million total supply distributed as follows: 50 million to stakeholders, 30 million to users of the service, 12 million reserved for rewards, promotion, marketing, etc.; the remainder funds network development and operational costs.

3.7.5 Storj

Storj is the decentralized cloud storage service. Unlike ordinary cloud storages, Storj is run on users' computers, who earn rewards in Storj internal currency, Storjcoin X (which runs on the Counterparty protocol), by providing disk space for the needs of the network.

3.7.6 Cuber

Cuber is a money transfer application using an approach similar to that of Tether. Digital assets in Cuber are claims against the project's partner bank (Estonian LHV Bank); the exchange rate of assets is fixed to Euro. Asset coin intricacies are largely hidden from users with the help of a mobile application (Cuber Wallet).

3.7.7 Koinify

Koinify was a marketplace for decentralized applications. The Koinify platform utilized Counterparty, but did not issue its own currency; to fund Bitcoin startups, Koinify sold application-specific tokens to be redeemed for application services. Koinify performed two major funding campaigns:

- GetGems, a social messaging service (see the description above)
- Factom (factom.org), an application for storing data on the Bitcoin blockchain (e.g., for proof of existence)

The Koinify platform was retired in May 2015 [66]. The developers claim to be working on a new platform that will incorporate emerging Bitcoin 2.0 technologies.

3.7.8 Swarm

Swarm was essentially a social network for cryptocurrency investors leveraged by the Counterparty platform. Swarm helped startups establish themselves as distributed collaborative organizations, allowing them to sell cryptographic tokens in order to raise funds. Swarm used its own currency, SWARM, for tokens. The platform was closed in September 2015 [67].

4 Conclusion

Blockchains could be one of transformative technologies for digital asset management, serving as a specialized platform as a service (PaaS) with significant growth potential. Blockchains could provide for unprecedented levels of counterfeit resistance, openness, transparency, and auditability. Blockchain technology could allow decoupling tasks associated with asset management and transaction processing, therefore providing an attractive alternative to existing centralized asset management

platforms for small and medium-sized businesses, third-party application developers and end customers. Internal, algorithmically enforced properties of blockchains (such as immutability) and their increased auditability could prove attractive for regulatory bodies.

Digital assets on blockchains could prove effective both in established financial services (e.g., for securities) and on emerging consumer-to-consumer and IoT markets. In the latter case, digital assets could be used in a variety of applications, including innovative financial services (e.g., crowdfunding, charity, peer-to-peer lending), smart property, digital subscription/access, and event tickets. Use of blockchains could facilitate management of assets by businesses, e.g., for discounts, gift cards, vouchers, and coupons. Blockchains could also prove effective in reducing the cost and expanding the reach of electronic money services for both currencies pegged to fiat money and alternative currencies.

In order to capture consumer-to-consumer markets and to maximize the scope of blockchain-based asset services in other cases, asset blockchains need to be open for third-party participation (including transaction processing, asset issuance and application development). Thus, permissionless or loosely regulated public permissioned blockchains could provide a fitting environment for customer-centric assets. Business-oriented assets could require blockchains with restricted access in the short term due to compliance. However, the transition of these assets to fully public blockchains is not out of question, as it would provide global reach, openness for innovation and better experience for end users. Disintermediation between parties – one of design principles of blockchain technology – could necessitate a transition towards the bearer paradigm of digital asset ownership, which would be augmented by third-party wallet services and legally recognized blockchain-based public key infrastructure.

The existing deployment models for blockchain-based digital assets include overlay protocols (i.e., colored coins and metacoins) and blockchains with native support of user-defined assets. The latter approach is inherently more scalable and could be more fitting for digital assets with higher transaction velocity or throughput, such as electronic money. However, existing multi-asset blockchains are not sufficiently tested and explored by cryptographers and scientists. Asset overlay protocols using the Bitcoin blockchain (Counterparty, OmniLayer, Open Assets Protocol, Colored Coins Protocol, ChromaWay, and CoinSpark), while not as scalable, utilize the well-tested infrastructure. The security and immutability properties of assets managed with an overlay protocol strictly equal to those of the underlying blockchain, thus making overlay protocols fitting for low-velocity assets with increased security requirements (e.g., smart property).

Public multi-asset blockchains and overlay asset protocols could form the basis for the IoV – a global, ubiquitous, largely permissionless network for digital asset transfer. While the technologies for this hypothetical network are not yet mature and the operation of the network poses unsolved regulatory and legal challenges and obstacles, blockchains could transform asset transfer in the same way the Internet has transformed data transfer.

Appendix A Blockchain Public Key Infrastructure

As blockchains provide a consensus state of a system with time-ordered atomic changes specified by transactions, blockchain infrastructure could be effectively used for creating a decentralized PKI [68, 69]. The PKI could be a part of the protocol for a digital asset blockchain, implemented as an overlay protocol (i.e., similarly to colored coin / metacoin protocols), or as a dedicated blockchain. A blockchain PKI protocol could specify the format of specialized transactions for basic tasks performed by network participants, such as creating, updating and revoking certificates for end customers, services, registration and certificate authorities.

As an example, consider the following procedure for creating a certificate for an end user or a service, which binds his identity to the public key P. We assume that the scripting capabilities of the underlying blockchain are similar to those of Bitcoin. For the sake of simplicity, we also assume that the certificate authority (CA) performs user authentication (i.e., this task is not delegated to a separate registration authority).

1. The user sends the payment for a certificate into an output, which could be redeemed by a 2-of-2 multisignature requiring both his signature (using the private key corresponding to P) and the signature of the certificate authority. Alternatively, the money could be refunded back to the user with a certain delay (e.g., three days) if CA does not act upon his request.
2. The user supplies CA with documents necessary for his authentication or the authentication of the corresponding company, together with a public key P and a reference to the payment transaction created on the previous step. The data may be submitted through a secure connection and digitally signed by the private key corresponding to P.
3. CA performs user authentication and creates a transaction **Tx** spending the user's payment. The spending transaction is encoded according to the blockchain PKI protocol and contains data on user's identity (the certificate). Certificate fields corresponding to sensitive information, such as a name or an address for physical persons, may be blinded and committed in the form of a hash; for services, the certificate could be fully disclosed. CA retains a copy of blinded fields in a secure storage, e.g., for law enforcement.
4. CA sends the half-signed transaction **Tx** back to the user together with additional information (e.g., a nonce used for blinded certificate fields). The user checks the validity of the certificate, signs **Tx** and broadcasts it into the blockchain.
5. Upon seeing the transaction **Tx** on the blockchain, network nodes acknowledging the corresponding PKI protocol check that it is co-signed by a valid certificate authority and satisfies other conditions. If **Tx** is valid, the corresponding certificate is associated with P.

Instead of writing certificate data directly into the blockchain, it could be committed to using a hash and served using a distributed hash table storage maintained by certificate authorities.

A certified public key could be used for asset issuance or as a receiving address for payments. In the latter case, the pay-to-contract protocol [22] could be utilized in order to create publicly unlinkable

addresses for payments, which would include information about purchase and permit audits. If the certified public key of the service is P (a point on the elliptic curve), then the payment address is (with minor simplifications) hP, where

$$h = \text{hash}(\langle \text{payment information} \rangle).$$

A user wanting to prove he has made a purchase could disclose h and the payment transaction.

Similarly, end users may utilize hierarchical deterministic wallets [21] with a certified public key high in the hierarchy (e.g., at the account level) in order to make and receive publicly unlinkable payments. A user wanting to prove he has made or received a payment, can do so by disclosing the appropriate path in the public key hierarchy.

Consider a situation where the receiver of the payment wants to verify the identity of the sender before the payment is made (e.g., for the KYC process). The sender does not broadcast the signed transaction, but rather sends it to the receiver via a secure channel together with the public key P, which has an attached certificate. Note that P does not coincide with the public key the sender used for the payment P_s. The receiver wants to ensure that there exists a link between P and P_s. The sender may provide the explicit link between these public keys or use interactive or non-interactive zero knowledge proofs.

As an example, consider labeled wallets described in [22]. In this case, the link between public keys is $P_s = rP$, where r is a random positive integer not exceeding the order of the elliptic curve[7]. The sender may simply send r to the receiver, or they could use a simple interactive zero-knowledge proof procedure:

1. The sender sends P and $P_s = rP$ to the receiver.
2. The receiver selects a random integer q and calculates $P_c = qP$. The receiver then sends P_c to the sender.
3. The sender calculates rP_c and sends its hash to the receiver.
4. The receiver checks if the received value is equal to $\text{hash}(rqP) = \text{hash}(qP_s)$. If it is, the receiver is now sure that the sender knows r. This, together with the fact that the sender knows the private key corresponding to P_s (witnessed by the payment transaction) ensures the identity of the payee.

If the sender uses a cryptographically secure random number generator, the knowledge of r cannot help the receiver in discovering other public keys utilized by the sender. r and P could be stored by the receiver and used in audits to prove identities of senders. Due to the nature of zero-knowledge proofs, they could only be useful if auditors place some trust in the receiver.

After the receiver has verified the payee's identity, he may proceed by broadcasting the payment transaction. Optionally, the receiver may prepend his signature of the transaction to the unlocking

[7] Note that labeled wallets imply plausible deniability in the event the discrete logarithm problem for elliptic curves is efficiently solved, e.g., with the advent of quantum computers. In this case, a certified public key could be linked to *any* public key ever used on the blockchain.

script of the payment transaction (it does not change the validity of the transaction). This could signal that the transaction is properly verified by the receiver. In this case, if the transaction contains a change output, the sender should add an additional signature himself in order to conceal the change. Note that signing the transaction puts the payment receiver at a greater risk, as he needs to keep his private key readily available.

The above procedure could be adapted for the case the payee uses several outputs for payment and/or multisignatures (e.g., if the payee uses a non-custodial multisignature wallet).

A more advanced alternative to labeled wallets and the pay-to-contract protocol is reusable payment codes for hierarchical deterministic wallets [70]. Compared to the scheme described above, reusable payment codes offer several advantages:

- As payment codes use HD wallets, the complete transaction history could be recovered from a single seed. In contrast, labeled wallets require storing additional data
- Identification data is exchanged using on-chain notification transactions. These transactions allow for easier independent audits

Another Bitcoin innovation related to PKI is the Bitcoin payment protocol [71]. The protocol binds requests of Bitcoin payments, which are issued by merchants, to the well-established X.509 PKI [72]. (X.509 certificates are used, in particular, to establish secure Internet connections via HTTPS). In our opinion, the payment protocol, while undoubtedly useful for creating trust in the Bitcoin ecosystem, could be insufficient by itself to create the complete blockchain-based PKI:

- The protocol binds X.509 certificates to payment requests rather than Bitcoin addresses. A request to pay to any Bitcoin address could be signed by any certificate. (That is, a payment request with a certain Bitcoin address cannot be used as a proof that an address is controlled by the signing party)
- The private key of a certificate needs to be readily available in order to sign payment requests, which could create vulnerabilities
- The X.509 PKI by itself is unrelated to blockchains (most X.509 certificates are issued to check authenticity and provide end-to-end encryption for websites). Meanwhile, blockchain PKI could benefit from domain-specific certificates; e.g., a bank authority could grant a certificate to issue assets, and government agencies could issue certificates for businesses
- The issuance and revocation of X.509 certificates is not audited by the blockchain, while blockchain technology provides an opportunity for such audits
- The payment protocol could be of limited use for peer-to-peer payments and for customer identification (e.g., according to the KYC procedure)

Yet another approach to blockchain PKI is used in IBM Open Blockchain [73]. Certificate authorities in Open Blockchain issue two kinds of transaction-related certificates:

- Long-term **enrollment certificates** linked to identity of their owner such as a physical person, a service provider or a validating node
- Short-term pseudonymous **transaction certificates**, the linkage of which to real-world identities could be requested from CA by proper authorities (e.g., by an auditor or by law enforcement)

Compared to approaches described above, Open Blockchain uses PKI more widely, as identities of transaction signers are determined by certificates instead of simple public keys.

Blockchain-based public key infrastructure requires further discussion on core design principles:

- What is the best deployment model for a PKI protocol? Could it be a universally supported overlay protocol (cf. with HTTPS) or a part of the blockchain specification itself?
- How could a PKI protocol adapt to discrepancies in national legal requirements?
- What roles and commands are needed for a PKI protocol? Could, e.g., courts be granted with special kinds of certificates, which would allow revoking third-party certificates or freezing funds?
- Could a PKI protocol utilize a web of trust approach with self-signed certificates linked to external platforms as per [29] (e.g., SSL/TLS-secured website locations for companies; social networks for physical persons)?

While these questions remain presently unanswered, a universally accepted blockchain-based PKI could become one of the integral parts of the IoV (cf. with use of certificates in Internet applications).

Appendix B Asset Coin Use Cases

We illustrate some of the possible use cases for digital asset coins with diagrams resembling UML sequence diagrams [74], which are commonly used in computer science. Examples use Counterparty as an asset protocol; with minimal changes, they can be adapted for other protocols.

- Figure 2 shows how asset coins can represent shares
- Figure 3 depicts dividend payments in a shares use case
- Figure 4 provides an example of using asset coins for voting
- On Figure 5, digital asset coins are used to represent discount
- Figure 6 depicts the use asset coins as access tokens

The diagrams do not detail the operation of the asset management platform, which may lead to an incorrect line of thought that the operation of blockchain-based asset management platforms does not differ from that of centralized platforms. In reality, main differences between the two approaches lie in the non-functional aspects (e.g., blockchain-based platforms do not have a single point of failure, are distributed, permit disintermediation between clients, etc.), whereas functional aspects may indeed be similar up to a certain degree of precision.

In all figure captions, we use phrases like “Alice creates a transaction” for the sake of simplicity. Most described operations could be automated with the appropriate software and performed behind the scenes, so end users would not need to deal with intricacies of asset protocols.

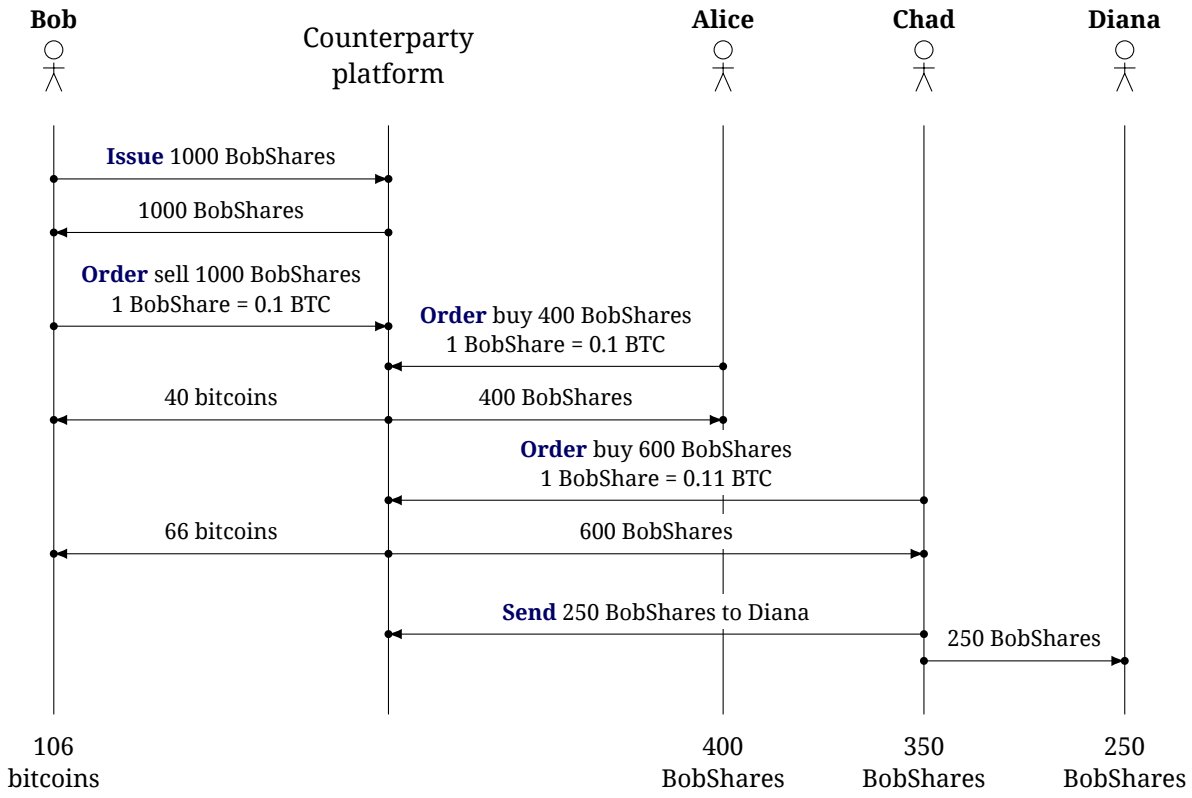

Figure 2: Using asset coins on the Counterparty platform to represent shares. In this example, Bob issues 1000 shares to crowdfund his new project and publicly sells them using a sell order. Orders in Counterparty can be executed in multiple parts. In this example, there are two buyers: Alice and Chad. Chad owes Diana some money; she agrees to take a portion of Chad’s BobShares as a payment.

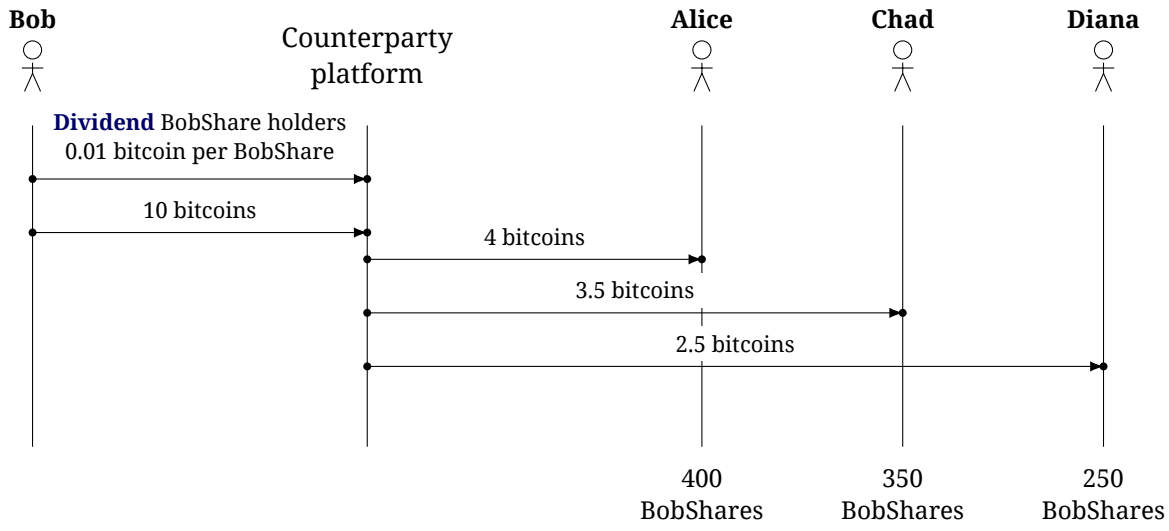

Figure 3: Using Counterparty’s **dividend** message (continued from Fig. 2). Bob distributes 10 bitcoins among holders of BitShares proportionally to their holdings.

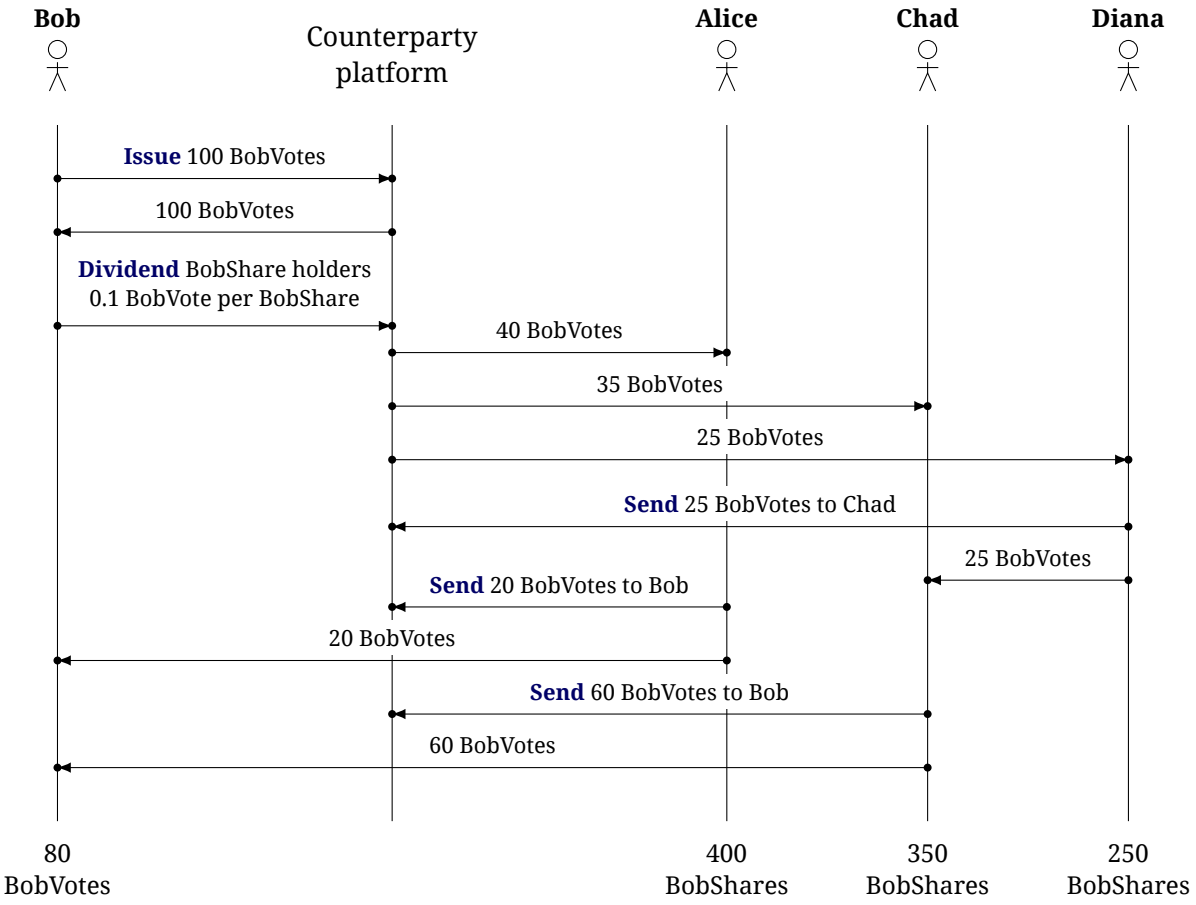

Figure 4: Voting implemented with digital asset coins (continued from Fig. 2). In this example, Bob decides to make some changes to his product and wants to ensure his investors support him in this decision. Bob issues new tokens (BobVotes) and distributes them among investors using **dividend** message; Bob declares that to make a change, at least 75% of BobVotes need to be returned to his address. Diana entrusts her decision on the matter to Chad by transferring ownership of her BobVotes. Chad supports the decision and sends his and Diana's votes to Bob's address. Alice is good whether or not the change is implemented; to indicate this, she sends *a half* of her votes to Bob's address. As 80 BobVotes (i.e., 80%) were sent back to his address, Bob now can safely implement the change.

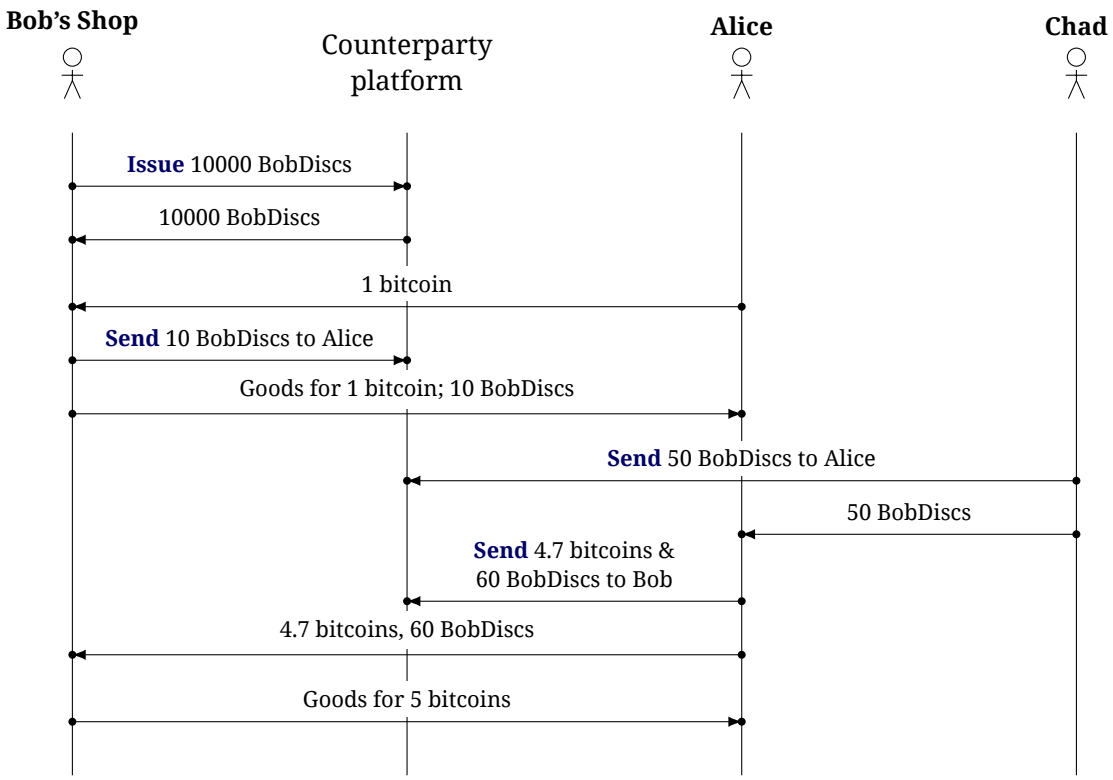

Figure 5: Using asset coins to represent discounts. Bob owns a shop accepting payments in bitcoins and wants to introduce a discount program. To accomplish this, Bob issues a large number of asset coins (BobDiscs). Each BobDisc represents a 0.1% discount and can be earned by buying goods for 0.1 bitcoins. Alice buys 1 bitcoin worth of goods at Bob's shop and receives 10 BobDiscs in return. Later, Chad gifts Alice 50 more BobDiscs. When Alice wants to buy 5 bitcoins worth of goods, she makes a transaction that includes 60 BitDiscs; this grants her $5 \cdot 6\% = 0.3$ bitcoin discount.

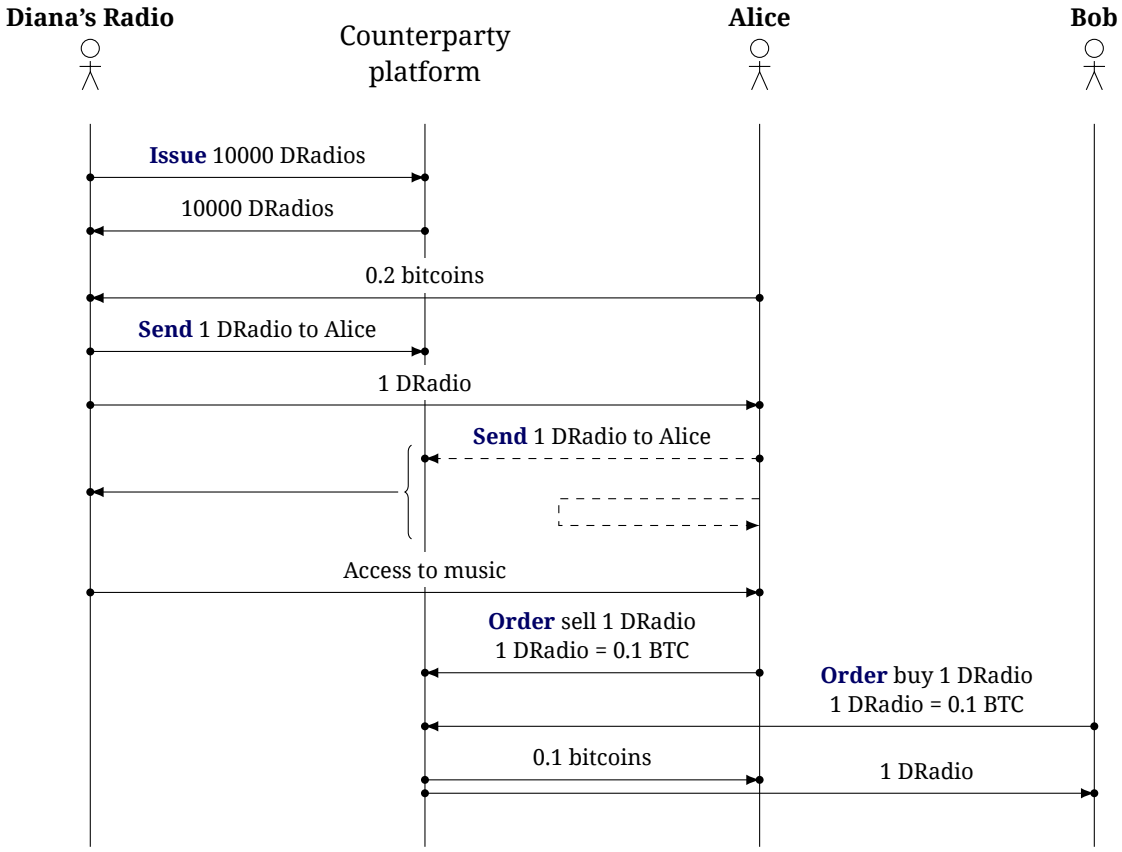

Figure 6: Using asset coins to grant access to digital content. Diana owns an Internet radio; she issues digital asset coins (DRadios), each of which grants access to the radio for a limited time period (e.g., 1 year since the token was issued). Alice buys a DRadio on the radio's website for 0.2 bitcoins soon after the tokens are issued. When Alice wants to listen to the radio, she creates a transaction transferring her DRadio back to herself and sends this transaction to the radio service as a proof of ownership (note that the transaction does not need to be broadcasted over the network). Diana then checks that the transaction is valid and its input is associated with a non-expired DRadio token. There are other ways to prove ownership of digital assets; e.g., Diana could challenge Alice with a random big integer number (nonce) and require Alice to digitally sign it with her private key. Alice can freely sell DRadio on any asset coin exchange. The buyer can specify the buying price that corresponds to the time until DRadio expires. Suppose Alice sells her token 6 months after it was produced (i.e., halfway through the expiration term); DRadio price by then should be near a half of the initial price.

References

[1] Blockchain vs./and Bitcoin with Adam Ludwin. In: a16z Podcast.
URL: http://a16z.com/2015/11/11/blockchain-bitcoin-fintech/

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

[3] *BitFury Group* (2015). Public versus private blockchains. Part 1: permissioned blockchains
URL: http://bitfury.com/content/5-white-papers-research/public-vs-private-pt1-1.pdf

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

[5] Sybil attack. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Sybil_attack

[6] *Tim Swanson* (2015). Permissioned distributed ledgers
URL: http://www.ofnumbers.com/wp-content/uploads/2015/04/Permissioned-distributed-ledgers.pdf

[7] *Gideon Greenspan* (2015). MultiChain private blockchain
URL: http://www.multichain.com/download/MultiChain-White-Paper.pdf

[8] Overview of Openchain. In: Openchain Documentation
URL: https://docs.openchain.org/en/latest/general/overview.html

[9] *BitFury Group* (2015). Proof of stake versus proof of work
URL: http://bitfury.com/content/5-white-papers-research/pos-vs-pow-1.0.2.pdf

[10] Colored coins. In: Bitcoin Wiki
URL: https://en.bitcoin.it/wiki/Colored_Coins

[11] *Meni Rosenfeld* (2012). Overview of colored coins
URL: https://bitcoil.co.il/BitcoinX.pdf

[12] *Alex Mizrahi* (2015). A blockchain-based property ownership recording system
URL: http://chromaway.com/papers/A-blockchain-based-property-registry.pdf

[13] *Tim Swanson* (2015). Watermarked tokens and pseudonymity on public blockchains
URL: http://r3cev.com/s/Watermarked-tokens-and-pseudonymity-on-public-blockchains-Swanson.pdf

[14] Electronic money. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Electronic_money

[15] Platform as a service. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Platform_as_a_service

[16] Separation of concerns. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Separation_of_concerns

[17] *Pieter Wuille* (2015). Segregated witness (segwit) and deploying it for Bitcoin. In: Scaling Bitcoin Hong Kong
URL: http://diyhpl.us/wiki/transcripts/scalingbitcoin/hong-kong/segregated-witness-and-its-impact-on-scalability/

[18] *Ethan Heilman, Alison Kendler, Aviv Zohar, Sharon Goldberg* (2015). Eclipse attacks on Bitcoin's peer-to-peer network
URL: https://eprint.iacr.org/2015/263.pdf

[19] *European Banking Authority* (2015). Guidelines on Internet payments security
URL: http://www.eba.europa.eu/regulation-and-policy/consumer-protection-and-financial-innovation/guidelines-on-the-security-of-internet-payments

[20] *Vitalik Buterin* (2014). Bitcoin multisig wallet: the future of Bitcoin
URL: https://bitcoinmagazine.com/articles/multisig-future-bitcoin-1394686504

[21] *Pieter Wuille* (2013). Hierarchical deterministic wallets (BIP 32)
URL: https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki

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

[23] *Gregory Maxwell* (2015). Confidential transactions
URL: http://elementsproject.org/elements/confidential-transactions/

[24] Non-interactive zero-knowledge proof. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Non-interactive_zero-knowledge_proof

[25] Secure multi-party computation. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Secure_multi-party_computation

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

[27] *Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green et al.* (2014). Zerocash: decentralized anonymous payments from Bitcoin (extended version)
URL: http://zerocash-project.org/media/pdf/zerocash-extended-20140518.pdf

[28] Estonian ID card. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Estonian_ID_card

[29] *ColoredCoins* (2015). Digital asset verification
URL: https://docs.google.com/document/d/1NYEYGI7oCRCjtbxRTLXes7M_-PeOwBF_SmwgmV1z78k/edit

[30] Merged mining specification. In: Bitcoin Wiki
URL: https://en.bitcoin.it/wiki/Merged_mining_specification

[31] Next-gen colored coin client base
URL: https://github.com/chromaway/ngcccbase/

[32] *Flavien Charlon* (2013). Open Assets Protocol (OAP/1.0)
URL: https://github.com/OpenAssets/open-assets-protocol/blob/master/specification.mediawiki

[33] The Colored Coins Protocol
URL: https://github.com/Colored-Coins/Colored-Coins-Protocol-Specification/

[34] Omni protocol specification (formerly Mastercoin)
URL: https://github.com/OmniLayer/spec

[35] Protocol specification. Counterparty
URL: http://counterparty.io/docs/protocol_specification/

[36] CoinSpark for developers
URL: http://coinspark.org/developers/

[37] Elements Alpha
URL: https://github.com/ElementsProject/elements/tree/alpha-0.10-multi-asset

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

[39] *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

[40] *Alex Van de Sande* (2015). Ethereum in practice part 1: how to build your own cryptocurrency without touching a line of code. In: Ethereum Blog
URL: https://blog.ethereum.org/2015/12/03/how-to-build-your-own-cryptocurrency/

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

[42] *Sergio Demian Lerner* (2015). RSK: Bitcoin powered smart contracts
URL: https://uploads.strikinglycdn.com/files/90847694-70f0-4668-ba7f-dd0c6b0b00a1/RootstockWhitePaperv9-Overview.pdf

[43] *Grace Caffyn* (2015). Everledger brings blockchain tech to fight against diamond theft. In: CoinDesk
URL: http://www.coindesk.com/everledger-blockchain-tech-fight-diamond-theft/

[44] *Yessi Bello Perez* (2015). How Provenance is channeling the blockchain for social good. In: CoinDesk
URL: http://www.coindesk.com/provenance-channeling-blockchain-social-good/

[45] *Pascal Paillier* (1999). Public-key cryptosystems based on composite degree residuosity classes. In: Advances in Cryptology – EUROCRYPT '99, LNCS vol. 1592, pp. 223–238.
URL: http://link.springer.com/chapter/10.1007%2F3-540-48910-X_16

[46] Script. In: Bitcoin Wiki
URL: https://en.bitcoin.it/wiki/Script

[47] Stack machine. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Stack_machine

[48] Transaction. In: Bitcoin Wiki
URL: https://en.bitcoin.it/wiki/Transaction

[49] *Andreas M. Antonopoulos* (2014). Mastering Bitcoin: unlocking digital cryptocurrencies. O'Reilly Media, 298 p. Chapter 4: Keys, addresses, wallets
URL: http://chimera.labs.oreilly.com/books/1234000001802/ch04.html

[50] Data storage methods. In: Colored Coins Protocol Specification
URL: https://github.com/Colored-Coins/Colored-Coins-Protocol-Specification/wiki/Data%20Storage%20Methods

[51] Multisignature. In: Bitcoin Wiki
URL: https://en.bitcoin.it/wiki/Multisignature

[52] (2015) Bitcoin Core 0.11.0 release notes
URL: https://github.com/bitcoin/bitcoin/blob/master/doc/release-notes/release-notes-0.11.0.md

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

[54] Two-phase commit protocol. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/Two-phase_commit_protocol

[55] *Pete Rizzo* (2015). Blockstream to launch first sidechain for Bitcoin exchanges
URL: http://www.coindesk.com/blockstream-commercial-sidechain-bitcoin-exchanges/

[56] Our conversation with Alex Mizrahi, CTO of ChromaWay

[57] *Andrew Miller* (2012). Storing UTXOs in a balanced Merkle tree (zero-trust nodes with $O(1)$-storage). In: BitcoinTalk Forums
URL: https://bitcointalk.org/index.php?topic=101734.0

[58] EPOBC: Enhanced padded order-based coloring. In: Next-gen colored coin client base wiki
URL: https://github.com/chromaway/ngcccbase/wiki/EPOBC

[59] ITOG parameterized color kernel. In: Next-gen colored coin client base wiki
URL: https://github.com/chromaway/ngcccbase/wiki/Itog

[60] transaction.h. In: Bitcoin Core Github repository (retrieved on Sep 14, 2015)
URL: https://github.com/bitcoin/bitcoin/blob/master/src/primitives/transaction.h

[61] Why proof-of-burn. Counterparty
URL: http://counterparty.io/news/why-proof-of-burn/

[62] 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/

[63] Counterparty community update, Aug 17th: Coindaddy asset vending machine service, Counterparty smart contract system & more. In: Counterparty News
URL: http://counterparty.io/news/counterparty-community-update-aug-17/

[64] CoinSpark assets
URL: http://coinspark.org/developers/assets-introduction/

[65] *Pete Rizzo* (2015). Hands on with Linq, Nasdaq's Private Markets blockchain project. In: CoinDesk
URL: http://www.coindesk.com/hands-on-with-linq-nasdaqs-private-markets-blockchain-project/

[66] *Tom Ding* (2015). Something on the verge. In: Koinify Blog
URL: https://koinify.com/blog/something-on-the-verge/

[67] *Joseph Young* (2015). Swarm shuts down as 'pretty boy' co-founder blamed for demise
URL: http://cointelegraph.com/news/115218/swarm-shuts-down-as-pretty-boy-co-founder-blamed-for-demise

[68] EmerCoin
URL: http://emercoin.com/

[69] Guardtime
URL: https://guardtime.com/

[70] *Justus Ranvier* (2015). Reusable payment codes for hierarchical deterministic wallets (BIP 47)
URL: https://github.com/bitcoin/bips/blob/master/bip-0047.mediawiki

[71] *Gavin Andresen, Mike Hearn* (2013). Payment protocol (BIP 70)
URL: https://github.com/bitcoin/bips/blob/master/bip-0070.mediawiki

[72] X.509. In: English Wikipedia
URL: https://en.wikipedia.org/wiki/X.509

[73] *IBM* (2016). Open Blockchain protocol specification
URL: https://github.com/openblockchain/obc-docs/blob/master/protocol-spec.md

[74] UML sequence diagrams. UML Diagrams
URL: http://www.uml-diagrams.org/sequence-diagrams.html