Covalence: Building sharing economies on an immutable ledger

Tom Trevethan, Nicholas Gregory and Lawrence Deacon
CommerceBlock
tom@commerceblock.com

Abstract

The current push for decentralisation has produced a variety of innovative applications, and created a new vision for how to structure social and economic networks. However, while showing strong potential, full decentralisation of control does not cater for the natural and necessary need for ownership that is fundamental to many businesses and organisations. Who owns the network? Who is responsible for the network? Open and distributed networks with centralised policy and legal accountability can provide a balance between decentralisation, transparency, governance and ownership that is scalable, efficient and secure. This paper introduces *Covalence* - a new model and philosophy for designing and building distributed sharing token economies.

Concept

Networks, whether human orientated or software driven, all require an essential common component: participation. From established networks such as SETI-at-home [1] and Uber [2], we have seen how an application can achieve a substantial network effect very rapidly, by freely and openly enabling users to join the network and participate. Rather than building a network from scratch, like in the case of cloud based infrastructure, where every component must be deployed by the network owner, work and services can be outsourced: SETI-at-home, Uber and AirBnb [3] are prime examples of this philosophy working in practice. This approach enables large and valuable networks to be established quickly, creating new economic models and opportunities, with lower risk and lower initial outlay [4]. The Covalence model utilises the concept of a token based economy to coordinate open and rapidly deployable networks, adopting the notion of a *network token*. In this system, an open but permissioned blockchain platform provides transparency and immutability for coordinating participation and tracking the behaviour and performance of the network participants.

In this new model, access to the network and permission to participate (i.e. to engage in the provision of a service in exchange for a payment) is administered by a central entity - the *coordinator* - but in such a way that all network participants can independently view and verify the global state of network participation which is governed by consensus rules. This is achieved via a fully public and immutable blockchain - the so-called *service chain* - where the agreements required for participants to provide services to clients is coordinated, recorded and tracked [5]. The service chain defines ownership of a *network token* (for the CommerceBlock network this is CBT) which is used to both enable and assign value to network participation. The service chain then embodies a public but *permissioned* ledger - which is operated by the coordinator [6]. This gives the coordinator ownership and legal responsibility over the network, however every action and transaction that occurs is immutable and must follow the codified consensus rules, as well as being explicitly visible to every network participant [7].

Client networks

The network tokens that coordinate service access and incentivise participant behaviour on the service chain are freely tradable and derive a market value that reflects the value, utility and security of the network and in turn the potential for token holders to generate profit via service provision. As participants are incentivised to participate in the network, the value of tokens will increase. To access and participate in the network (i.e. to provide services in exchange for payment), token holders are required to stake or lock tokens (that become illiquid for some period of time) in order to obtain a *ticket* which then enables the participant to provide specific services to clients. Service chain network participants (via a ticket) can then provide services to separate but linked networks known as *client networks*. These linked networks utilise the services provided by the service chain participants. This enables client networks to rapidly and efficiently acquire the distributed network effect of the service network in a secure and verifiable way [8].

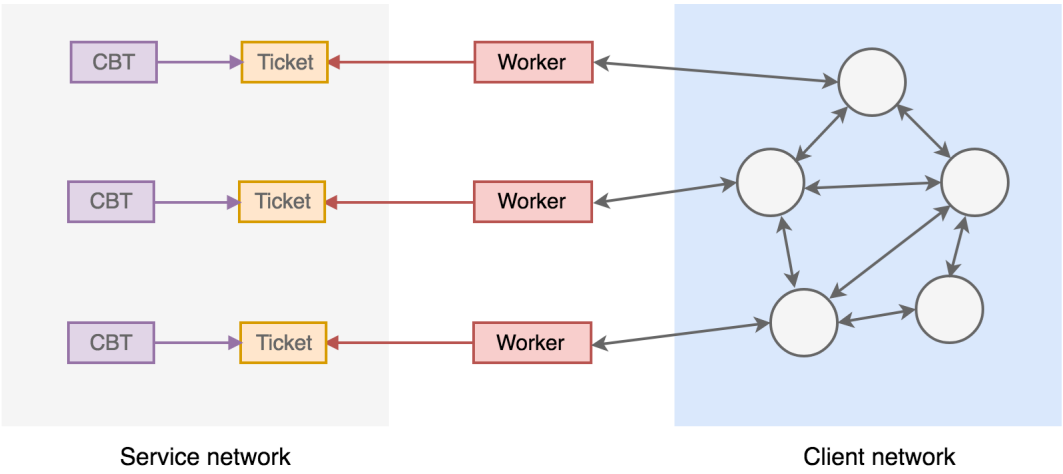

Figure 1: Schematic of a client network interacting with service chain participants offering services as 'workers' via tickets.

Client networks can *connect* to the service chain in order to obtain the services of the network participants. The client network will typically also be a permissioned system (i.e. a separate permissioned blockchain) administered by an entity referred to as *commissioner*. The commissioner has the authority to offer a proportion of the resources or income generated from the client network operation (e.g. transaction fees) to *workers* - participants of the service network who have obtained permission to provide services to the patron network via the possession of a ticket. This enables the client network to rapidly acquire network effects, which is critical to their growth and security.

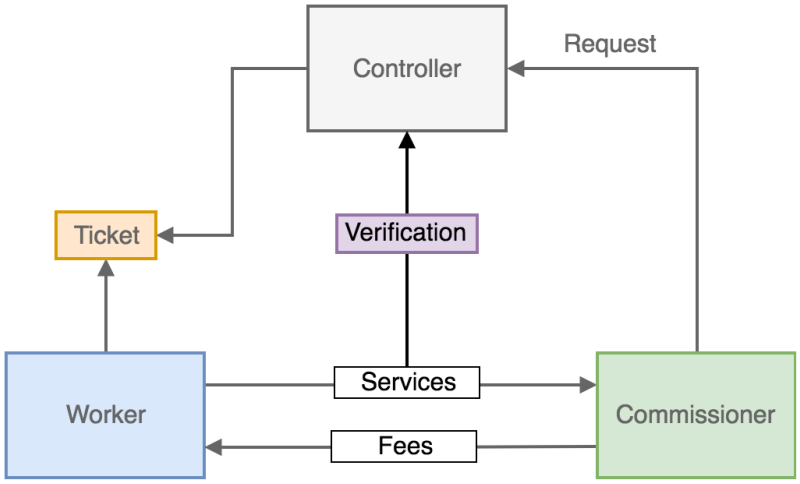

Figure 2: Schematic of the relationship between workers, the service chain coordinator and the client network commissioner.

Requests and service agreements

In the process of establishing the service agreement, the commissioner of a client network determines the proportion of network fees (or other resources) that will be paid to workers, the services the workers are required to provide and the number of workers required (this number determines the target number of tickets issued). This information is encoded in a request that is submitted to the service chain, which then creates the specified tickets and is publicly viewable by all service network participants (both via service chain network nodes/wallets and service listing web-based explorers). This request is effectively a 'contract for work', and it consists of rules for ticketing parameters, defined service parameters and compensation terms. Permission is required from the service chain coordinator in order to create a request.

The service network participants can then evaluate the request and make a decision on whether to stake network tokens required to obtain a ticket, dependent on their own evaluation of the incurred costs in providing the requested services. The amount of tokens required to be staked (or locked) is determined by the demand for the tickets of a particular request, which in turn will be related to the potential profit for the ticket holder (worker) and the opportunity cost of locking tokens (which will depend both on the network token value and volatility of the price).

Any number of commissioners for separate client networks can create requests, and benefit from the distributed services provided by the root network participants. As the value of the services increases the value of the network goes up, further increasing the demand for tokens and increasing the reliability of network participants. This gives the network tokens value that is retained - and aligns the incentives in the system to increase network utility.

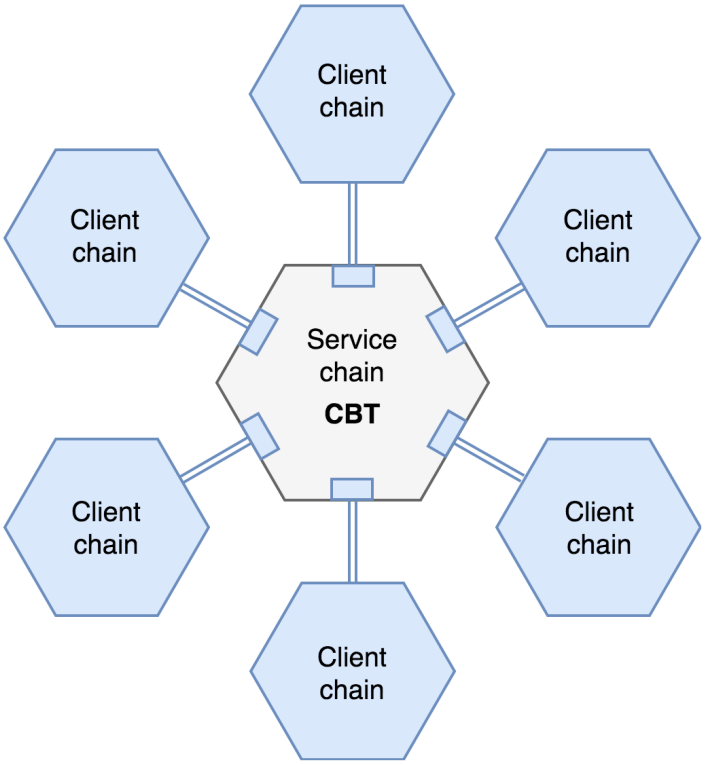

Figure 3: Illustration of the linking of multiple client networks to the service chain via requests.

Requests and service agreements

The services provided by the workers can in principle be anything that the client network requires, that can be reliably measured and verified. In the case that the client network is operated as a public blockchain, these services will typically be functions that increase the security, resilience, reliability, performance and decentralisation of the client network (or client blockchain).

These services can include:

- Distributed and replicated storage of data (archival blockchain storage).
- Client network (consensus) rule enforcement and monitoring.
- SPV proof generation for client network wallets.
- Fraud proof generation and propagation for client network nodes.

Workers must then provide these services continuously, and the work must be verified by the commissioner (via tools and infrastructure provided by the service chain coordinator). Work-

ers automatically generate *proofs of service* (such as proof of blockchain storage, challenge-response queries etc) which are coordinated via the service chain and made available to commissioners and client chain networks.

After verification of services provided (over the periods specified in the request) the workers (ticket holders) then receive the proportion of client network fees specified in the request. The request can then be renewed (automatically if necessary) and the cycle repeated. In addition, the ticket holders (workers) of a particular request are awarded *reputation tokens* dependent on the reliability of the services they provided during the specified period (as determined via the service verification). These reputation tokens can then be used to reduce the quantity of network tokens that required to be staked to obtain new tickets in subsequent requests. By demonstrating the ability to offer service reliability and consistency, reputation tokens are accumulated, further decreasing the network tokens required for a specific ticket.

Incentives and token model

The Covalence model is built on the notion of an open network, where anybody is free to participate: this is a central requirement in order to be able to rapidly grow network effect and scale. Therefore, the network participants are relatively free, and cannot be directly controlled or coerced. However, the consensus rules of the service chain and network token can be structured in such a way to incentivise behaviour which optimises the objective function of the system, which is an emergent property of the actions of individual network participants [9]. The Covalence model objective function is: **to maximise the value provided to client networks by service chain participants (workers)**. This is then implicitly linked with the maximisation of both the income (fees) paid to the workers, and the maximisation of the value of the service chain network token.

To maximise this objective function, service network participants must be incentivised to perform the required services reliably and consistently [10]. The incentivisation is driven primarily by 1) the maximisation of the payment received (client network fees) and 2) by the minimisation of the tokens required to be staked to obtain a ticket. In order to achieve the first of these, the client network commissioner must determine the minimum number of service providers necessary to meet their requirements, which is set as the target number of tickets: n_p. This number determines by how many ways the total payment (F_p) will be split, with each ticket holder (i) receiving:

$$I_i = \frac{F_p}{n_p}$$

Therefore, as n_p is increased, the incentives to both obtain a ticket and perform services decreases. Restricting n_p to the minimum requirements for the client network increases demand for tickets, and increases the ticket price, which in turn increases the incentives for workers to maintain reliability. As the ticket price increases, participants are further incentivised to maximise the discount they can obtain via the acquisition of reputation

tokens (RTs) - which are awarded (issued) after a worker successfully completes a specified service agreement as verified by the coordinator. Each worker i begins with a reputation token $R_i = 0$, and this is incremented at the completion of each service interval $R_i \leftarrow R_i + 1$.

The incentive for acquiring reputation tokens is that their possession reduces the quantity of network tokens required to be staked (locked) for the acquisition of a ticket from a particular request. The network tokens required to be staked for a ticket is determined through a *uniform price Dutch auction* algorithm (as described below) which determines an auction price A_p for the tickets of request p. The number of network tokens then required to be locked for a ticket is then:

$$T_p = \frac{A_p}{\log(CR_i) + 1}$$

where the dependence on the reputation score follows an inverse logarithmic function [10]. The parameter C can be set (in the client request) to modify the dependence and optimise for specific network behaviours. This function ensures a continuous benefit to providing consistent and reliable service, while not significantly preventing new network participants (without any reputation tokens) from obtaining tickets - ensuring open access to the network.

Ticket auction mechanism

To create stable incentives for workers to provide reliable and consistent services, the access to client networks must be artificially restricted by allocating only a limited number or tickets. In order to allocate tickets to workers fairly and transparently, network participants are required to stake network tokens (i.e. lock them for a specified period). Therefore the market value of the network tokens will reflect the emergent value of the services provided by network participants (and the potential for profit via fee income). In order to allocate tickets for a particular request p in a way that ensures optimum participation as well as efficient valuation, the network token staking requirement is determined via a uniform price Dutch auction algorithm [11]. This enables efficient stake price discovery - each network participant can create a bid for any amount of network tokens at what they judge to be the correct value of the ticket [12]. Each bid is visible and all subsequent bidders benefit from the bid information. Once the auction algorithm has finalised, winning bidders are allocated tickets for the stake requirement, which is determined by the auction algorithm, irrespective of the bid amount.

This auction protocol starts with a very high stake price ($S_p(0)$) which then continuously decreases as a function of time t, based on a predefined function (that converges to zero as $t \to \infty$). Once the auction is initiated (on the submission and confirmation of a request with ticket parameters), network token holders submit stake commitments at the instantaneous stake price $S_p(t)$ which bind them to the request service agreement. As the stake price decreases and more network participants enter bids, when the number of bid commitments reaches the target ticket number n_p the auction finalises (at time $t = t_f$). At this point

the final stake price $A_p = S_p(t_f)$ is applied to each ticket stake independent of the specific bid amount - each participant stakes the same amount of network token A_p (or $T_p(R_i)$ if discounted by the participant reputation token R_i). This stake is then locked (in a time-locked on chain smart contract) for the duration specified in the request service agreement. At the end of this period, the participant is free again to transact (sell) the stake tokens.

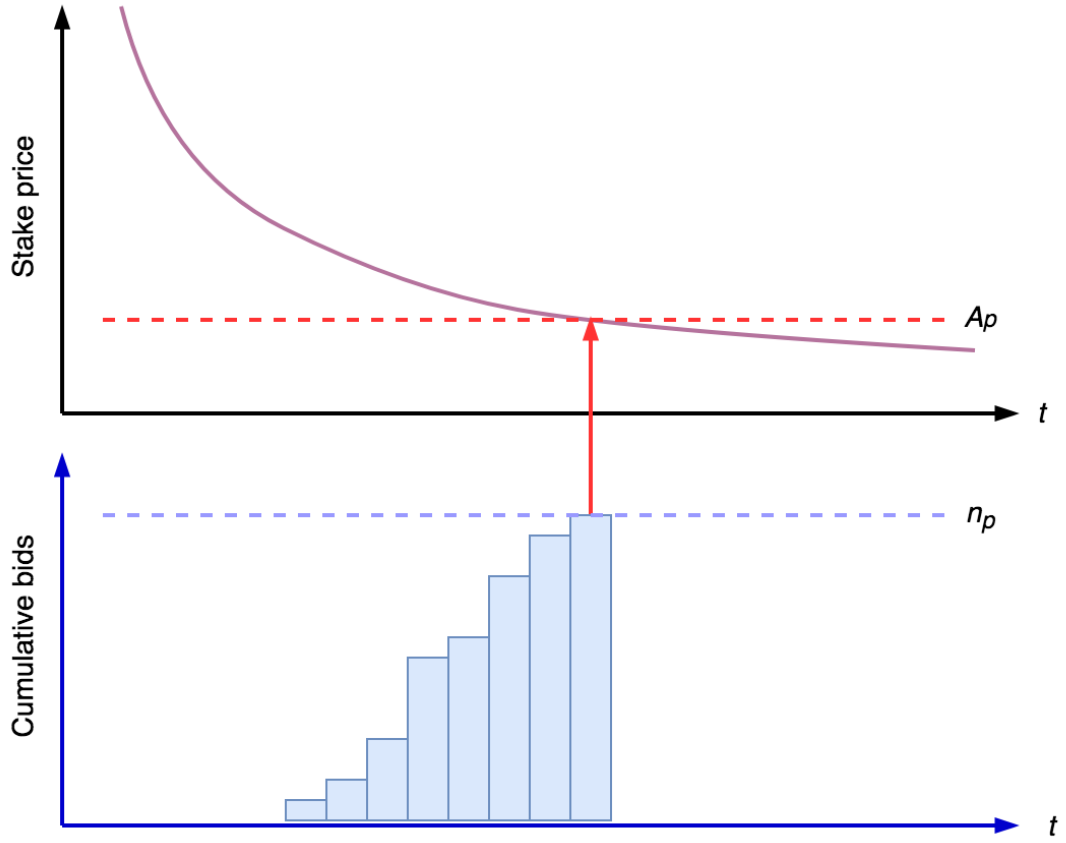

Figure 4: The upper curve shows the stake price decrease over time. The lower curve shows the cumulative number of bids. When the number of bids reaches n_p the current ticket stake is set.

This mechanism enables rapid and efficient discovery of the ticket stake price [13], and all successful ticket bidders obtain tickets at the same stake. Collusion between bidders to lower the final stake price is impossible to eliminate, however the effect of this is minimised by ensuring the bidding process is transparent and visible to all participants. The stake remains time-locked for the duration of the service contract, and never leaves the custody of the ticket holder. Automated tooling and wallet can enable a seamless continuation and renewal of service contracts without user intervention.

Conclusion

The model presented in this paper elucidates a new approach to creating, valuing and rewarding network participation that enables rapid deployment and growth of distributed applications. By emphasising the ownership and accountability of network coordinators in addition to transparency and immutability, this model provides a platform that fits much better with the requirements of modern enterprises than fully decentralised and permissionless systems. By providing the tools and incentives for anyone to participate and provide services to a growing ecosystem of networks, this platform can create a new economies based on participation and transparency.

[1] Korpela, Eric, et al. "SETI@ HOME—massively distributed computing for SETI." Computing in science engineering 3.1 (2001): 78-83.
[2] Cramer, Judd, and Alan B. Krueger. "Disruptive change in the taxi business: The case of Uber." American Economic Review 106.5 (2016): 177-82.
[3] Zervas, Georgios, Davide Proserpio, and John W. Byers. "The rise of the sharing economy: Estimating the impact of Airbnb on the hotel industry." Journal of Marketing Research 54.5 (2017): 687-705.
[4] Puschmann, Thomas, and Rainer Alt. "Sharing economy." Business Information Systems Engineering 58.1 (2016): 93-99.
[5] Swanson, Tim. "Consensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems." Report, available online, Apr (2015).
[6] Xu, Xiwei, et al. "A taxonomy of blockchain-based systems for architecture design." Software Architecture (ICSA), 2017 IEEE International Conference on. IEEE, 2017.
[7] Puthal, Deepak, et al. "The blockchain as a decentralized security framework." IEEE Consum. Electron. Mag. 7.2 (2018): 18-21.
[8] Alabi, Ken. "Digital blockchain networks appear to be following Metcalfe's Law." Electronic Commerce Research and Applications 24 (2017): 23-29.
[9] Trent McConaghy, "Token Design as Optimization Design", 9984 Blockchain Meetup, Feb. 7, 2018, Berlin, Germany, https://www.slideshare.net/TrentMcConaghy/token-design-as-optimization-design
[10] Ocean protocol https://oceanprotocol.com/tech-whitepaper.pdf
[11] https://medium.com/@raiden$_n$*etwork*/*the* − *raiden* − *token* − *auction* − *explained* − 1*cc*0*c*7946*b*26
[12] Vickrey, William. "Counterspeculation, auctions, and competitive sealed tenders." The Journal of finance 16.1 (1961): 8-37.
[13] Smith, Vernon L. "Auctions." Allocation, Information and Markets. Palgrave Macmillan, London, 1989. 39-53.