

The Credit Protocol Whitepaper v1.0.1

Blockmason Team

2017-08-15

Introduction & Features:

There are three ways to consider a new Ethereum-based protocol: the value it creates for a user, the value it creates for a developer, and the value it creates for the community. At Blockmason, we are dedicated not only to building useful, functional applications and protocols that solve real-world problems, but also to developing platforms that inspire adoption of the Ethereum network and expand global access to cryptocurrency.

That is why we are so excited to announce the development of our newest product: **The Credit Protocol** (CP).

Until now, it was only possible to move money on the blockchain in the form of cash. Bitcoin democratized the transfer and storage of money, and Ethereum democratized the creation and storage of monetary contracts. With the Credit Protocol, BlockMason has taken the next logical step in the decentralized economy: democratizing the creation of credit. Debt and credit are already extremely powerful financial tools, and now they will be strengthened by the security and flexibility of the blockchain. In fact, because an individual need not own ether in order to draw debt in ether, it is entirely possible to imagine a future in which debts and credits recorded on the Credit Protocol could exceed the total market capitalization of all cryptocurrencies combined, dramatically increasing the scale of the digital economy.

It is entirely possible to imagine a future in which debts and credits recorded on the Credit Protocol could exceed the total market capitalization of all cryptocurrencies combined, dramatically increasing the scale of the digital economy.



However, there is no better proof of a team’s vision and execution than its product. At BlockMason, we have never set much store by empty promises or flashy Whitepapers. Yes, we are extremely proud of the work our team has put forth to create this document, but we are even more proud of our developers, who have already deployed a fully-functional beta version of our newest application, Friend In Debt, demonstrating the functionality of the Credit Protocol. This awesome, simple DApp is available for public testing *right now*. Go ahead, check it out at [fiddyio](#), we’ll wait...

Oh, you’re back. Well, before you read about our new protocol, how about one last thing:

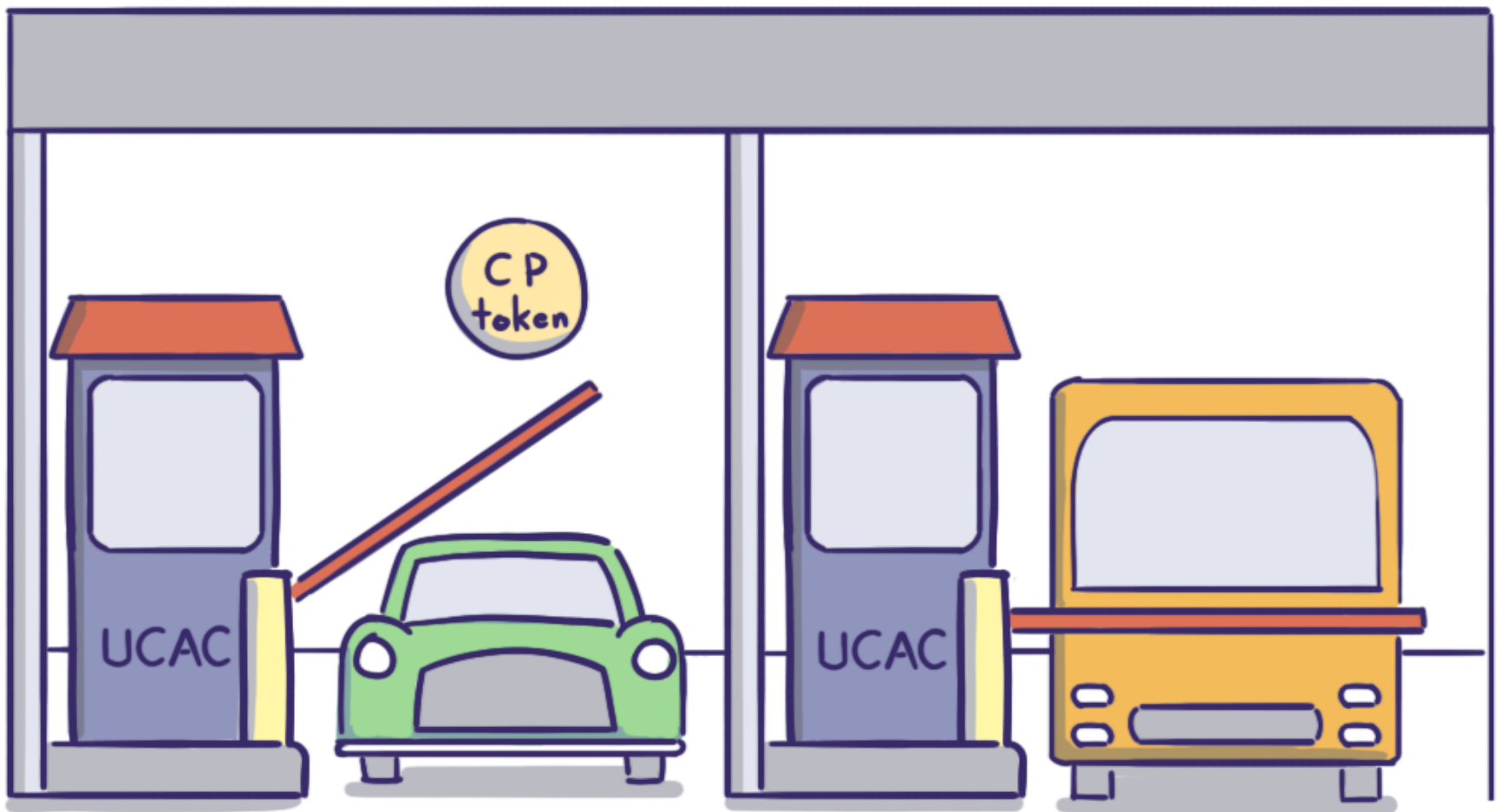
When purchasing product-use tokens, it is important to not simply look at the product as is, but also consider the company’s ability to provide exemplary customer support for that product. That is why we will continue to update and improve CP in advance of and throughout the duration of the token. Unlike other sales, based on hope and potential, our work stands alone to demonstrate our company’s commitment to ongoing development, product maintenance, and creativity as we expand the possibilities of blockchain technology. Our passion for our product and desire to improve the Ethereum ecosystem is not dependent on funding, and we aim to prove that we love building useful, functional, life-changing DApps—no matter what.

The Credit Protocol

At its most basic level, the Credit Protocol is a system for recording debts and credit between entities on the Ethereum blockchain. One entity sends a debt or credit request to another, and that user then confirms the debt or credit, which is recorded and stored within the Credit Protocol’s smart contract.

This simple interaction between entities via the Credit Protocol enables a deceptively powerful and complex array of possible transactions when coupled with a Use Case Authority Contract (UCAC) built atop the protocol. Each UCAC has its own set of rules about what types of debts it will record, when it will record them, and from whom it will accept them. For example, a UCAC may permit transactions from only certain certified users, allowing organizations to build private debt recording contracts. These contracts could be used for anything from tracking accounts receivable to developing private organizational currency, as is used on many college campuses.

In many ways, this system is similar to a highway. Each debt or credit is akin to a car—there are different makes, models, features, and colors, but they all serve a fundamentally similar purpose. Once a car is allowed onto the highway, this debt, and all its particularities, is recorded. But, of course, you may desire some regulation to keep all that traffic running smoothly and going to the right place. Enterprising developers can construct their own tollbooths, or UCACs, at the entrance to the highway. Each tollbooth has complete control over what types of cars it allows on to its particular stretch of highway—sports cars or semi-trucks or huge vans filled with puppies—and when those cars may access the highway. Of course, operating a tollbooth requires resources. Users of the CP will have opportunity to purchase a Credit Protocol Token (CPT), which funds the movement of cars through the tollbooth and onto the highway. Users may charge a fee to allow cars to travel the highway, or they may let the cars pass for free.



The CP

The amount of cars allowed through a tollbooth depends on two factors: how many CPTs are funding the tollbooth's operation, and how many transactions are permitted per CPT. An algorithm will adjust the number of transactions permitted per one CPT based on network traffic, meaning that transaction costs are dynamic and adjustable based on the needs of the network.

Through these CPTs, a UCAC effectively buys debt recording capacity from the network. CPTs are, in essence, a license to use the CP network, much as software is licensed to users. This license is granted in perpetuity to the holder of a CPT for as long as he or she owns that token. The owner of a CPT may choose to “sublicense” their allotted transactions to another user, whether for free or for compensation, while retaining their token and the rights granted therein.

Use Case Authority Contracts

Use Case Authority Contracts act as a pathway for inputting and settling debts. Because UCACs are user-created, the particular rules that govern recording debt through a UCAC may be optimized to best fit the need of the organization or user employing the Credit Protocol. Parameters for which a UCAC may be programmed include currency type, the size and frequency of allowable transactions, which users may create debt or be extended credit, limits of debt accrued, and much more. The possible iterations of effective, valuable UCACs is restricted only by the imagination of the designer, so long as its result is a debt transaction compatible with the Credit Protocol.

Each UCAC requires the operator to “**stake**” a minimum of one CPT to write debt or credit through that UCAC to the Credit Protocol system. Each CPT possessed by the operator permits the UCAC to process a specified number of transactions per day, a number continuously and dynamically determined by an algorithm within the CP smart contract. Therefore, staking more CPTs allows for greater transaction throughput within the UCAC. Depending on the intended debt recording system, users may need to stake multiple CPTs to

UCAC. Depending on the intended debting system, a user or developer may need to own multiple CPs to optimally guarantee the UCAC throughput capacity. While users may own fractional tokens, these partial tokens cannot be used to generate transactions. The transactions generated by a single CPT expire 24 hours after generation, at which point the CPT will generate a new set of available transactions.

Bounty Program for UCACs

While we have spent significant time and energy conceptualizing useful UCACs that take advantage of the Credit Protocol to solve real world problems, we know that we cannot envision every application of the CP alone. Because there are infinite variations of Use Case Authority Contracts, as well as varying demand for the creation of such contracts, the Credit Protocol token sale will set aside a portion of the CPT sales revenue to fund a Bounty Program that incentivizes developers to write high-demand UCACs that can show the way for developers to build apps.

Free UCAC Depository

At BlockMason, we are dedicated to providing outstanding customer support for the Credit Protocol and its users. A vital step toward that goal is building a supportive community that believes in the Credit Protocol and further developing its unique ecosystem. In that spirit, we will create and maintain a depository of free UCACs for developers to leverage in building their own DApps on top of the Credit Protocol. This depository will include all winners of our Bounty Program, as well as generic templates of successful UCACs that may be customized for specific applications. This depository will not only encourage the growth of a strong CP community, but it will also promote the adoption of the CP protocol by prospective users who can browse a list of successful UCACs already being implemented throughout the world.

FoundationIDs

On July 1st, BlockMason released our innovative Foundation protocol for free. ^[1] We developed Foundation to solve the problem of multiple wallet integration. We believe Foundation provides a solution to obstacles encountered by numerous application developers, and integration with Foundation is vital to the success of the Credit Protocol.

FoundationIDs create a single Master ID from multiple Ethereum public addresses, establishing a unified identity and transactional platform for individuals on the Ethereum Network. Users with FoundationIDs may easily sign into multiple DApps with one login credential (similar to how most web apps allow for sign in by Facebook or Google accounts) without the need for users to pass private keys back and forth. FoundationIDs will initially be free for all users of the CP. As previously discussed, this integration will improve user experience and lower the technical barrier to entry.

For further information, see [Foundation Technical](#) below.

Enforcing Debt Repayment

While the CP provides a valuable, multipurpose tool for recording debts and credit, it is important to note that individual users must still agree upon how best to enforce repayment of those debts. This enforcement may take many forms, and could involve both on-chain and off-chain solutions, including: binding legal contracts, collateral in the form of physical / digital assets, or social mechanisms such as blockchain credit reporting or business reviews.

Ultimately, users should not lend to entities in whom they doubt the ability to repay their debt, nor should users lend more money than they are comfortable losing.

Through the customization and personalization of UCACs, however, it is possible to design debt contracts that perfectly match the needs and individual users and businesses, rendering the question of debt enforcement moot. While we outline several possible use cases of the Credit Protocol below, it is worth considering two common debting scenarios:

1. Jesse owes his friend, Tim, a beer.

2. Jesse’s business, Jesse Corp., issued his customer, Tim, a gift card.

In scenario one, social pressure is sufficient to enforce repayment of the debt once it has been recorded on the Credit Protocol through the UCAC—no one wants to be Jesse’s friend if he won’t get you back for that sweet, refreshing, cold-as-the-Rockies brew.

In scenario two, economic pressure is sufficient to enforce repayment of the debt once it has been recorded —no one will shop at Jesse Corp. if the business won’t honor its gift cards.

In both cases, the specific UCACs that control the types of debt being recorded have built-in mechanisms for debt enforcement. In general, this type of enforcement is particularly effective for debting scenarios in which a large company or institution owes a smaller beneficiary for prepaid goods or services.

Because legal and social mechanisms already exist for enforcing debt repayment, the true power of the Credit Protocol rests in its ability to reliably record debts through double confirmation. These eliminates the need for the ceremony or bureaucracy currently currently for parties to trust a debt. Recording debts on the blockchain allows individuals and entities to tap into existing mechanisms for debt transactions without the red tape of banks, lawyers, and other trust institutions.

The Ethereum Conundrum

Because each debt/credit transaction recorded within the Credit Protocol still writes data to the Ethereum blockchain, executing transactions requires the payment of Ethereum to act as gas. While this may seem like a substantial roadblock for applications hoping to integrate the Credit Protocol, we at BlockMason believe in a future in which Ethereum and other digital currency is as common as cash today.

The Ethereum network sits on the verge of massive reductions in transaction cost as the network adds off-chain processing and other improvements, meaning that users will soon pay mere fractions of a penny to interact with such decentralized applications.^[2] The recent announcement of Plasma by Vitalik Buterin and Joseph Poon further justifies our faith that Ethereum will soon be capable of swiftly handling large amounts of microtransactions.^[3] It also proves that the Ethereum Foundation is committed to Ethereum not only as a currency but as a platform for consumer decentralized apps.

In addition, BlockMason has already begun researching our own proprietary off-chain solution, to be implemented at the earliest possible date. Adapting to off-chain processing developed by others, or adopting our own version if it better suits the needs of the Credit Protocol, is part of our ongoing dedication to the maintenance and support of the CP. As interacting with blockchain applications becomes more commonplace, users will not even notice spending a few dollars a month on Ethereum transactions.

Furthermore, BlockMason is developing strategies for embedding Ethereum wallets within applications in order to pay for the transaction fees of new users, drawing potential customers from a new pool of clients not already familiar with cryptocurrency. For example, an application could automatically generate an Ethereum wallet for new users, and the company running that application could give a small amount of money to fund a user’s initial transactions as part of their user acquisition cost. In this way, such applications implementing the CP can push blockchain technology mainstream by lowering the technical barrier to entry.

1. https://blockmason.io/blog/2017/07/01/introducing_foundation/ ↔
2. <https://github.com/raiden-network/raiden/wiki/FAQ> ↔
3. <https://www.plasma.io/> ↔

Credit Protocol Usage Examples:

The simplest way to imagine the Credit Protocol is as a mechanism for the creation, tracking, and settlement of IOUs. At a basic level, IOUs are no different than currency. The earliest forms of cash were little more than standardized IOUs for some amount of precious metal, such as gold or silver, or a commodity like stored

grain. **Therefore, by allowing individual users to create and manipulate customized IOUs, CP effectively democratizes the digital issuance of currency.**

Therefore, by allowing individual users to create and manipulate customized IOUs, CP effectively democratizes the digital issuance of currency.

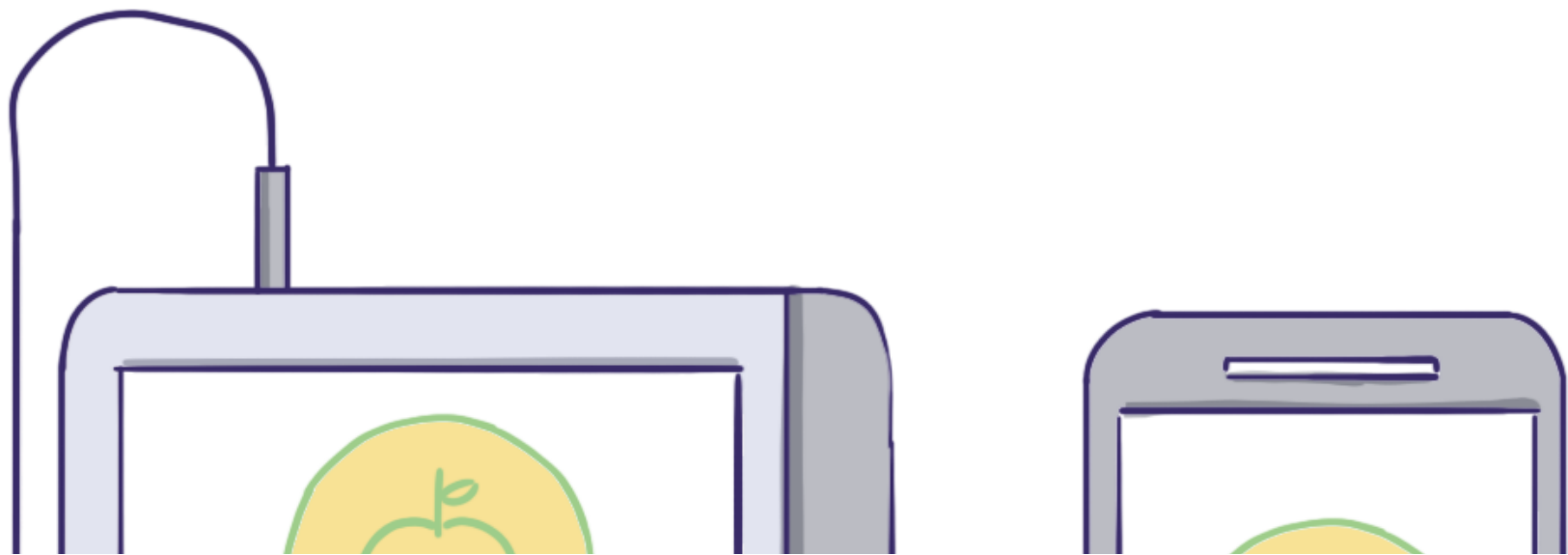
Off the blockchain, there already exist a massive range of use cases for IOUs—they exist between vendors and buyers in the form of accounts receivable, retailers and consumers in the form of gift cards, or airlines and flyers in the form of miles. Sometimes, these IOUs are denominated in a fiat currency issued by a government; other times, the IOUs exist only as an arbitrary currency designed by its issuer. Either type of IOU has value, so long as the two parties trust that one party may redeem the currency for something else of agreed upon worth. CP is the foundational tool that enables this massive range of use cases, available to and developable by anyone owning our product-use tokens.^[1]

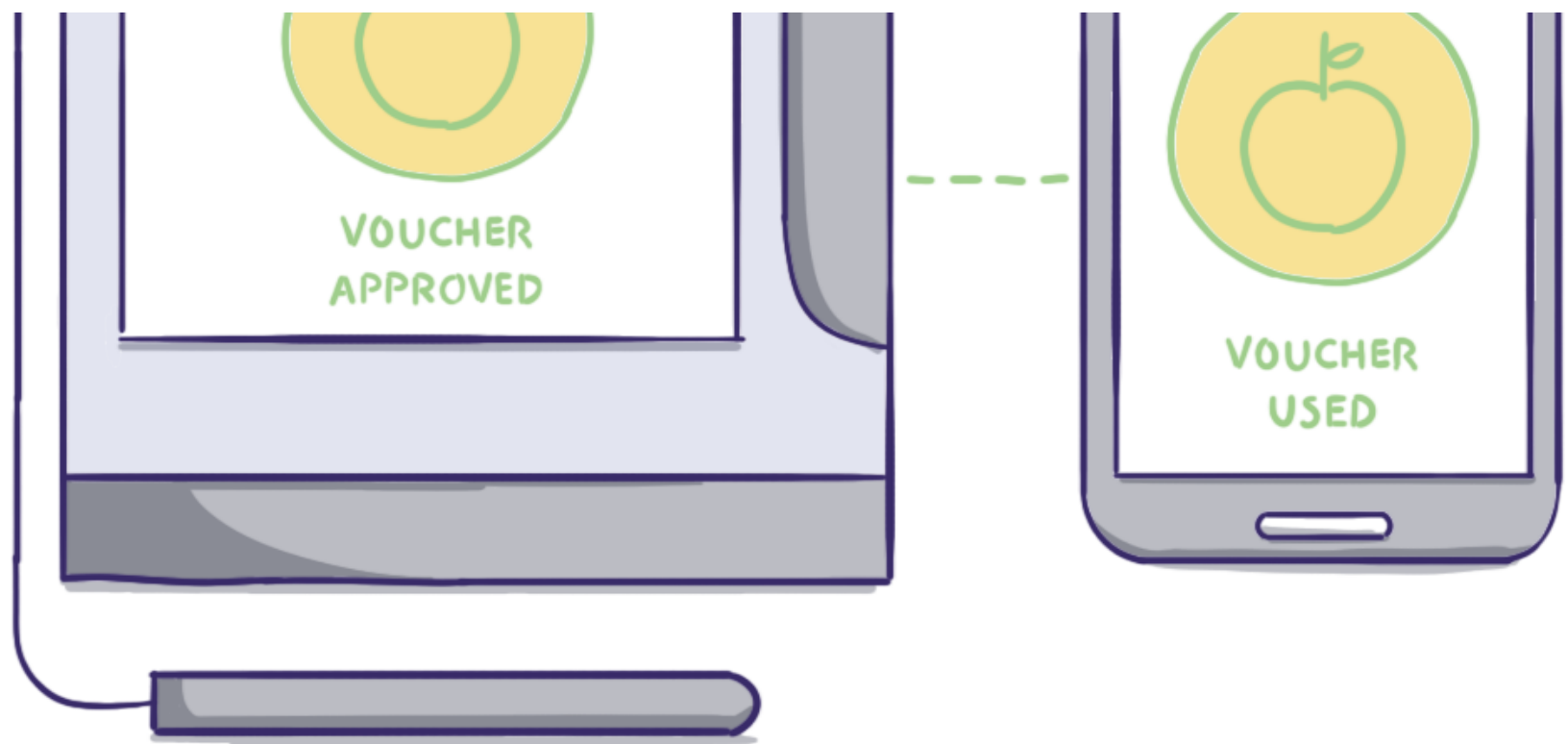
We envision a vast and diverse array of applications to be built atop BlockMason’s Credit Protocol, with tremendously varied functionality. Below, we examine just a few usage examples for the Credit Protocol—the first targets for our Bounty Program (see explanation below)—as well as our first exciting CP-based application: **Friend in Debt** (Try It Now).

We invite everyone interested in the Credit Protocol to check out our code at our github repository .

University Meal Vouchers

Imagine a school wants to implement a meal plan for its students. First, the school purchases sufficient CP tokens to cover the amount of transactions they expect to occur within the system. Then, the school whitelists students’ FoundationIDs, preventing anyone other than a university student from executing transactions through the UCAC. The UCAC is designed to only allow transactions between the school and students, preventing students from trading tokens between each other on any sort of underground market. The Meal Voucher system uses its allotted free CP transfers (proportional to the number of CP tokens purchased by the school) to allow students to settle debts without cost on the CP network. Students then pay for a meal plan, let’s say \$300 a month. Students pay the school in cash or digital currency, and the school sends the student 60 meal tokens. Each time the students visit the cafeteria, they tap their phone to machine that records their purchase, spending one meal token to buy their food. The school’s UCAC allows *only* these types of transactions to be settled between specific on-campus food venues and students of the school.





Microfinancing

A microloan system may be developed in the same manner as the meal voucher system described above. The creator of the Use Case Authority Contract, in this case the lender, could write a contract that limits the amount of individual debts per borrower, requires multiple borrowers for one loan (e.g. for individuals from the same community or village), makes the pool of borrowers jointly liable, and includes a built-in interest and payback schedule.

Microfinancing of this type is an extremely powerful tool, especially for communities without access to traditional banking. It enables liquid movement of debt and complex economic development previously unavailable to areas of the world outside the reach of large financial institutions. These developments could include small business development loans, low-interest payday loans, or crowdsourced repayment of community debts, such as rebuilding a business after a fire or paying medical expenses after an accident.

Retail Gift Cards

Another obvious use of the Credit Protocol is an application to manage retail gift cards. Consumers may purchase store credits and send those credits to any FoundationID (including his or her own), creating a debt between the retailer and the recipient of those credits. When the recipient uses his or her credits to redeem a product or service, it settles that debt.

Alternatively, a user could purchase from a retailer a redeemable code that allows access to funds already written into the Use Case Authority Contract. This digital ecosystem of debt and credit movement between retailers and consumers allows for the simple creation of entities like Visa Cash Cards, or all-purpose credits that can be deposited in a specific gift card account.

Because gift card usage is generally very similar between vendors, one can imagine a single unified application for managing any number of gift cards, allowing users to gift their friends credits which could then be assigned to specific stores of their choosing.

Consumer Loans

While securing a consumer loan currently requires wading through significant red tape and bureaucratic infrastructure, the Credit Protocol could vastly simplify the process of acquiring such a loan. Because blockchain ledgers are both public and secure, users could employ DApps to track their debts exactly, facilitate debt repayment through third party organizations, and be absolutely certain that payment has been received, avoiding any of the numerous scams or problems plaguing those operating in the debting world.

Such a system could also make use of the underlying debt data stored in the Credit Protocol to evaluate individual borrower's credit and debt repayment history. A UCAC could further verify a borrower's claims

manages server's credit and debt repayment history. It combines both rather than verifying a server's claims against information associated with that user's FoundationID to determine their legitimacy.

Get a suspicious phone call claiming you're behind on your car payments? Hang up and check the blockchain to confirm your payment *instantly*.

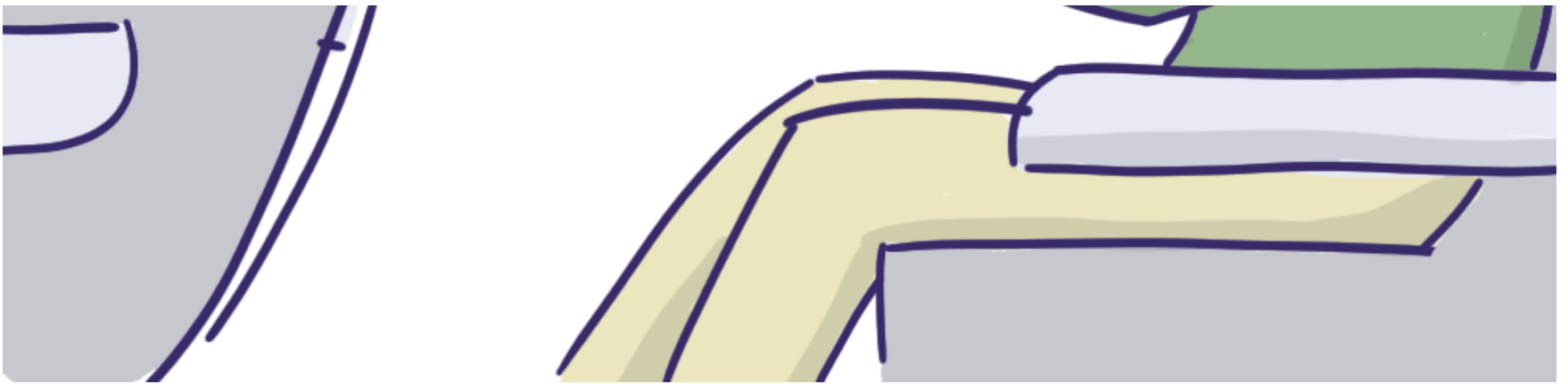
Airline Miles / Vendor Loyalty / Credit Card Points

Airline miles and vendor loyalty points are yet another form of debt that exists between the airline or vendor and their customers. Airlines issues miles in the form of their own currency (e.g. IOUs) as a reward for fliers based on their patronage. Users then settle the debt by claiming a product or service based on the point system established by the airline or vendor.

This vendor loyalty system highlights a particularly interesting property of the Credit Protocol: debts need not be recorded as fiat currency, such as USD, or cryptocurrency, such as ETH. Users may write UCACs to accept arbitrary values and currencies so long as:

- 1. they map redemption values to redemption methods
- 2. the customer base values the newly generated currency, such as airline miles or credit card points.





Here, we've saved the best for last.

Friend In Debt

Friend in Debt is our first full-fledged (**and functional!**) implementation of the Credit Protocol, a social tool for users to track and issue debts amongst friends, family members, and individuals in their community. Its potential applications and use cases are vast and powerful, and we look forward to presenting Friend in Debt in further detail later in this Whitepaper.

Or, to jump straight into the economic revolution, explore Friend in Debt here: fiddy.io

1. PLEASE NOTE: Some applications built upon CP may be subject to regulation in some jurisdictions. Developers building upon CP are responsible for determining which apps are subject to regulation and for complying as appropriate. ↔

Why CP?Security

Debting is an inevitable evolution in the blockchain ecosystem, and will grow only more prominent as individuals begin using Ethereum and other cryptocurrencies to finance the development of large-scale infrastructure. While individual solutions to handling digital debt will have inevitable insecurities, the Credit Protocol will be battle-tested and reliable. By employing the Credit Protocol, companies and individuals eliminate the need for backend engineers. All that is necessary is to design a Use Case Authority Contract and a front-end interface, with no need to worry about the security of your debts, credits, or assets.

By employing the Credit Protocol, companies and individuals eliminate the need for backend engineers. All that is necessary is to design a Use Case Authority Contract and a front-end interface, with no need to worry about the security of your debts, credits, or assets.

Ease

As the Credit Protocol develops its user base, the ease of creating contracts for CP will further increase as the community develops solutions to common problems and plans for common projects. Soon, there will exist template UCACs to address everyday financial issues faced by organizations large and small. In fact, we imagine an ecosystem similar to that of Web Page templates, in which a prospective user may search through potentially millions of different UCAC options, both free and paid.

Scale

While Bitcoin has created an entirely new economic ecosystem through its system of digital credits, the potential of crypto-economies is limited if they do not expand beyond the movement of currency. One of the most powerful factors in economic growth is debt and credit. The development of virtually all large-scale projects and infrastructure—bridges, factories, homes, technological innovations—are powered by debt. In

fact, the federal government of the United States, the largest economy of the world, would collapse overnight without the power to borrow and lend money. The ability to issue debt is a necessity for any significant growth in the digital economy.

The ability to issue debt is a necessity for any significant growth in the digital economy.

Additionally, it is worth noting that because recording one ETH of debt does not require owning one ETH, the total amount of debts and credits recorded on the Credit Protocol could soon exceed the total market capitalization of all cryptocurrencies combined. By opening CP to developers, we intend to maximize the speed of development, penetration, and usage of this democratized instrument for money creation.

Support

Working with new software is often a daunting task. For many, software support can mean the difference between days of struggle and immediate solutions. It can mean the difference between a failed venture and fully-functional, profitable application built atop the Credit Protocol.

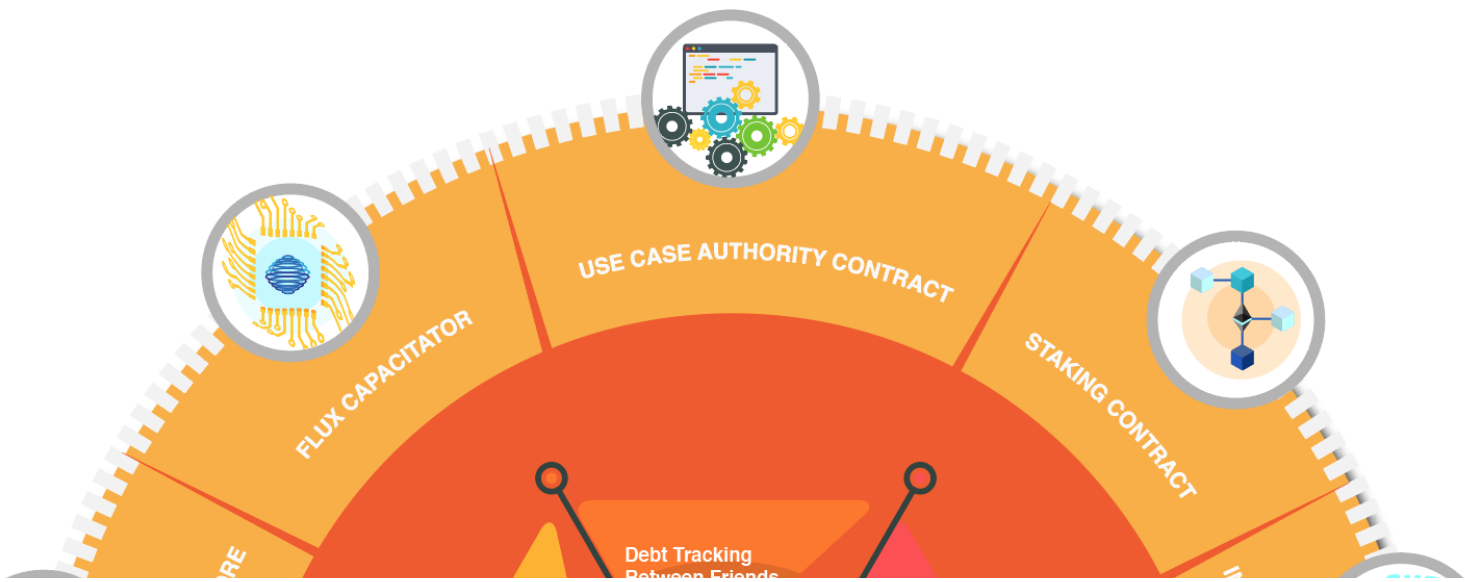
Through the Credit Protocol’s website, BlockMason intends to continue adding vital support resources for our customers and token holders. These resources will include detailed documents explaining how to maximize productivity when using the CP, how to adapt the CP for various use cases, and many examples and templates of successful projects. We will also provide API documents, model UCACs, and comprehensive front-end interfaces that seamlessly integrate with UCACs. In short, BlockMason will provide resources to guide customers from a nascent idea to a fully-realized UCAC and front-end interface built atop the CP.

Finally, at BlockMason, we are no entomophobics—we do not fear bugs; we destroy them. We take bug reports seriously, and will enable a dedicated bug reporting system for the Credit Protocol, allowing us to solve technical issues in a timely and efficient manner.

Urgency

The movement of debt and credit on the blockchain is vital to the growth of our ecosystem, and BlockMason has developed a protocol that works **right now**. Don’t wait around for a half-baked solution promised years down the line. We have already deployed a working version of CP, and we invite all prospective users to explore the code.

CP Technical



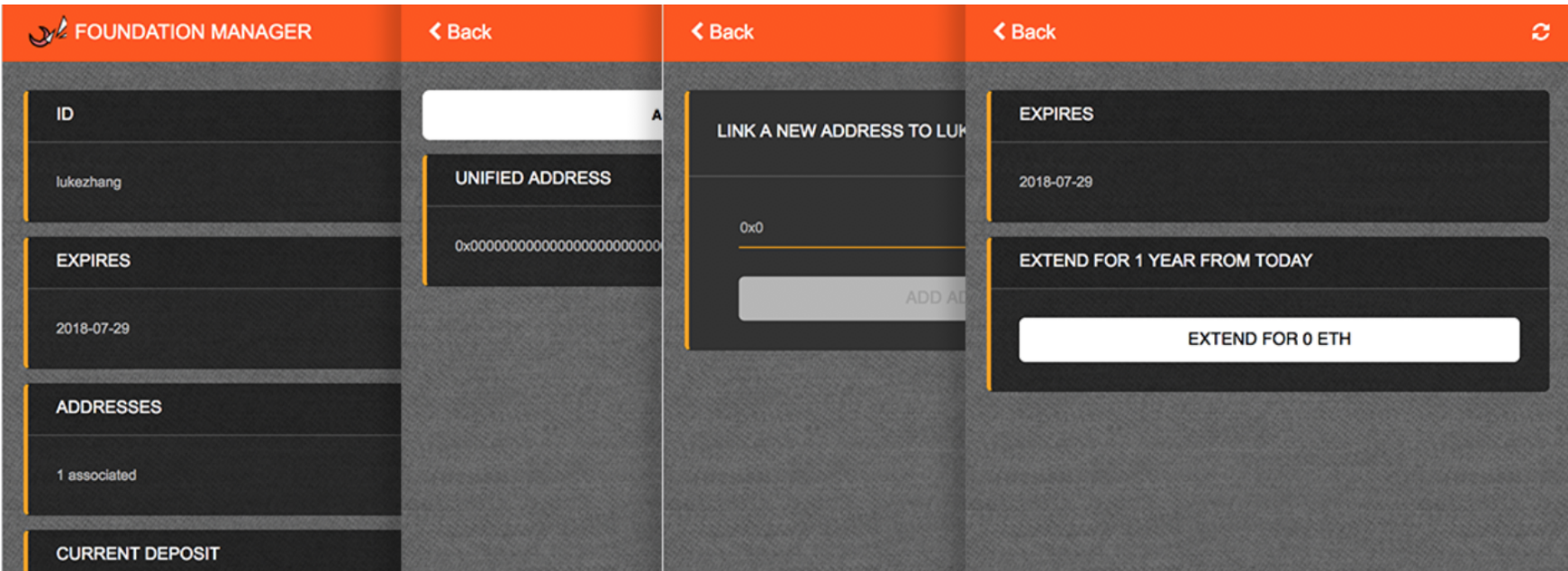


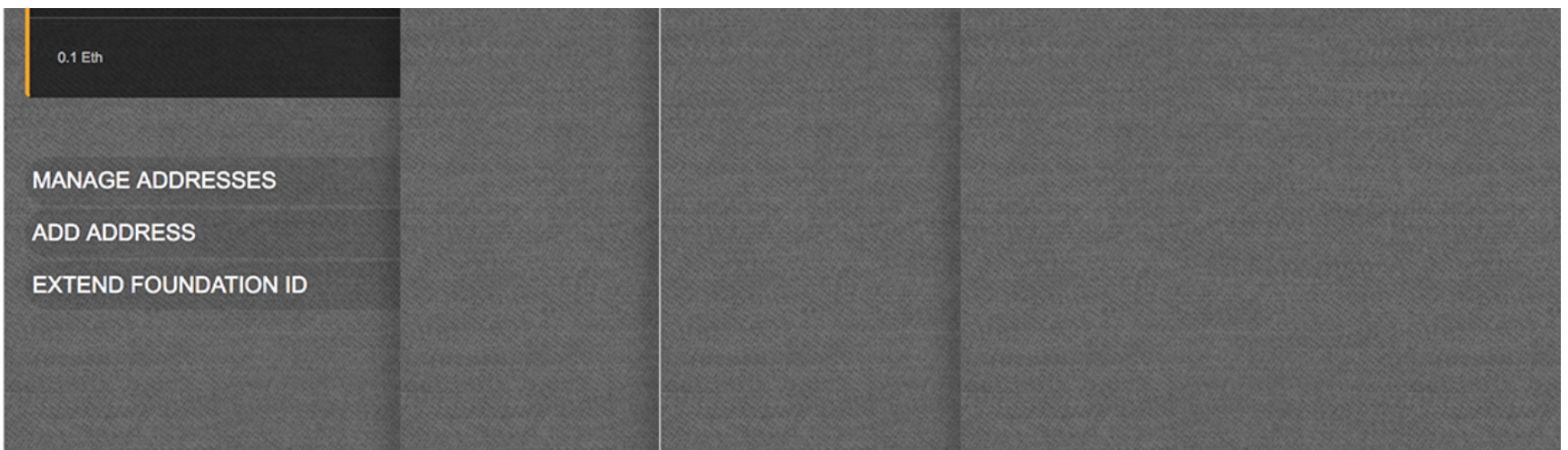
And for building on top of the Foundation Protocol, view our: [Foundation Technical Documentation](#)

Through endless coding, sleepless nights, and extensive trial and error, we realized that we required a unifying mechanism to link Ethereum addresses on one's phone and desktop. From the ashes of our earliest CP development, Foundation rose like a phoenix—a phoenix whose primary purpose is to allow all of your addresses to identify as **you**.

Foundation's interface facilitates a clean and easy ID creation process. All that is required is an Ethereum address not currently associated with another Foundation ID. It's that simple. Once a user has created a FoundationID, he or she may then choose to add any other addresses he or she currently owns from an address already associated with the Foundation ID, via the Foundation smart contract.

This unified FoundationID is required to use the Credit Protocol. It allows users to maintain one identity with the ability to access the CP from any device and any DApp browser.





Credit Protocol Data Smart Contract

The Credit Protocol Data smart contract is the core contract which stores data about all relationships between entities, as well as their debts and credits. It holds two main data structs: **Relationships** and **Debts**.

Relationships tracks relationships between two entities. To form a debt between two entities, those two entities must be demarcated as “connected.” Relationships also tracks pending relationships, as users are required to confirm relation requests.

Debts keeps track of debts between one user and another. It executes tasks like tracking debts, holding pending debts, and recording in which currency particular debts are denominated.

Credit Protocol Smart Contract

The Credit Protocol smart contract reads and manipulates the Credit Protocol data contract. This contracts functions include: reading user’s debts, pending debts, sending debts, confirming debts, reading friends lists, adding friends, removing friends, etc.

The Credit Protocol smart contract is the only address authorized to manipulate information stored in the Credit Protocol data smart contract.

Users may only interface with the Credit Protocol smart contract via the Flux Capacitor contract (*described below*).

Flux Capacitor Smart Contract

The Flux Capacitor contract is the contract with which users interact. The user never actually interacts directly with the UCACs. Flux Capacitor functions mimic the core CP contract functions, with the exception that each function also takes the address of a UCAC. The Flux Capacitor functions check the calls from the user against the UCAC at the UCAC address to make sure parameter values specified by the user are valid for that specific UCAC.

The Flux Capacitor contract also checks the Staking contract to record the amount of tokens a CP Token holder is staking and for which contracts those tokens are declared. The Flux Capacitor contract also notes total remaining capacity for transactions allowed within a given time period for that user’s staked tokens. The contract then updates the CP Token staker’s transaction capacity and calls the corresponding function in the CP Data smart contract.

It is easiest to think of the Flux Capacitor as an enforcer. It administrates the parameters of a UCAC and limits transaction capacity according to the amount of CP Tokens a user has staked for a given UCAC.

Use Case Authority Contracts

UCAC’s contain the exact same function calls as those listed in the Flux Capacitor contract. Each function

only returns a boolean of true or false. The Flux Capacitor contract calls on the UCAC with the exact function name as denoted in the Flux Capacitor to check if the transaction is valid.

As an example, the Flux Capacitor function `newDebt` would call a UCAC's `newDebt` function with the exact same parameters: `bytes32 myId`, `bytes32 friendId`, `bytes32 currencyCode`, `int amount`, `bytes32 _desc`, and the UCAC would return true or false to verify if this was a valid transaction. Therefore, if a UCAC allows EUR, but not USD, and a user attempts to pass USD through the Flux Capacitor contract, it would return false. A false return would cause the call to `newDebt` in the Flux Capacitor to fail.

A UCAC's reason for returning true or false answers are limitless. Their only requirement is that they maintain the same function names and parameters inputs as the functions in Flux Capacitor. If they do not have the same function names and parameters, then calls to those Flux Capacitor functions will throw.

Multiple Users and Bidding

Multiple users can stake CP tokens for the same UCAC, if allowed. If a DApp is sufficiently popular, hundreds or thousands of users may stake CP tokens to help support its transactions.

A UCAC may also be written to allow CP token holders to compete for transactions through that UCAC. For example, two users may stake their tokens for the same UCAC, offering their allotted transactions to users of the UCAC at two different prices. The DApp can then evaluate which bid is least expensive, and offer the price to the user before sending the CP transaction through that pathway.

Staking Smart Contract

The Staking contract stores information about how many tokens a user stakes to a specific UCAC, powering the transactions of a UCAC. When a user attempts to execute a function for a UCAC, the Flux Capacitor then queries the Staking contract for information about how many tokens the user has staked to power the transactions of that UCAC.

The Staking contract also watches for changes in a user's CP tokens to guarantee that the user is not trying to cheat the system to acquire free transactions.

Tech Considerations

Nameprep for FoundationIDs

FoundationIDs integrated with CP will be usable only if adhering to a nameprep standard that exclusively supports lowercase letters and numbers. This restriction means that FoundationIDs containing any symbol other than `lowercase a-z` and `0-9` will not be usable with the CP.

Storage Costs

Storing data on the Ethereum blockchain is expensive. In order to make CP low cost for users, BlockMason's ongoing maintenance and support include updating and optimizing our product to reduce costs. Cost minimization is an ongoing process that involves the use of systems like IPFS and other off-chain storage solutions. We carefully select variable types to minimize storage costs and also consciously examine the benefits of storing specific data on or off the blockchain.

Expandability

While we are already excited about the current version of CP, we are committed to providing robust maintenance and support, including fixes and expanded functionality in periodic version upgrades, as with any quality software-based service. With maintenance and support in mind, we have compartmentalized the Foundation smart contract to separate storage variables from actual functionality. This allows us to upgrade the protocol's functionality while maintaining all existing storage, saving the cost of porting that storage into new contracts. We will implement this same compartmentalization method for all CP smart contracts.

One powerful feature of building applications atop the Ethereum blockchain is its public ledger storage

One powerful feature of building applications atop the Ethereum blockchain is its public ledger storage. Because the CP smart contract is accessible to anyone on the Ethereum network, future developers may easily build applications atop the existing CP infrastructure. This transparency allows for the creation of dynamic new debt payment models using the existing CP data and functionality, including platforms for micro loans, consumer loans, and business to business credit lines.

Cost & Economics

When BlockMason designed the Credit Protocol, we envisioned a revolutionary protocol with global reach, available to users free of charge. However, computing, data storage, and transaction execution all have a cost on the Ethereum blockchain. This cost is commonly referred to as **gas**. Therefore, because CP runs on the Ethereum blockchain, using the protocol inevitably incurs transaction costs. Additionally, compared to traditional applications using standard web hosting, storage prices are relatively high. While we believe that Ethereum's global domination is inevitable, we also know that the average user need not necessarily understand Ethereum and its operation. Successful, scalable DApps will likely appear to users just as normal, web-based applications, with many layers between users and the Ethereum network.

The average user need not necessarily understand Ethereum and its operation. Successful, scalable DApps will likely appear to users just as normal, web-based applications, with many layers between users and the Ethereum network.

Plasma and Off-Chain Transactions

Right now, the entire blockchain community is grappling with the problem of balancing cost and network speed against security and longevity. Fortunately, it seems a number of improvements may be within our grasp. In particular, the recent announcement of Plasma ^[1] demonstrates the Ethereum Foundation's commitment to increasing network speed and decreasing transaction cost, paving the way for widespread adoption of consumer focused decentralized applications. Additionally, the Raiden Network, similar to Bitcoin's proposed Lightning Network, has the promise to reduce transaction costs up to 7 times and process up to 1 million transactions per second. It does this by moving some transactions off the blockchain onto a peer-to-peer network, while still interacting with the Ethereum Network to retain security and longevity. In the event that a Raiden Network like solution fails to be implemented for smart contracts, BlockMason will execute its own off-chain settlement feature and make it available to all users. Employing an off-chain solution allows CP to make fewer updates to the Ethereum blockchain, and thus reduce associated costs.

Most importantly, we believe that CP will be a significant factor in driving users to the Ethereum network. The protocol's power and its low barrier to entry will draw customers from a new customer base unfamiliar with cryptocurrency. We also plan to design future versions of the CP that cover the cost of a new user's transaction fees without their even needing to know about concepts like ether and gas.

CP will be a significant factor in driving users to the Ethereum network. The protocol's power and its low barrier to entry will draw customers from a new customer base unfamiliar with cryptocurrency.

UCACs, Transaction Costs, and Monopolies

CP token holders receive the ability to conduct a defined number of transactions per day based on the number of tokens they hold (*which can be fractional*). While holders may choose to employ their tokens in CP DApps they personally use, in order enable their own transactions, token holders may also pursue various developer pathways as well. For example, holders could stake a DApp that they control, ensuring their users have an existing pathway to process a defined number of transactions, either for a fee set by the holder or free of charge.

Token holders also might choose to stake their funds in other CP DApps to provide further transaction capacity for the DApp's users, whether for free or for compensation. For example, a UCAC may include a bidding system to encourage CP Token holders to provide pathways for their users. While these bidding systems could be written in a variety of ways, entirely up to the UCAC creator, one solution would be an open ETH marketplace in which the lowest cost transaction pathway would “win” and be “sold” to the user of the DApp.

Such a market need not necessarily evaluate pathways based on associated fees. Instead, software may choose pathways based on other factors such as the reputation of users, which could be determined by institutions completely unrelated to the Credit Protocol. For example, an environmental agency could certify on the blockchain that a CP token stakeholder has taken an action that had a positive or negative effect on the environment. The DApp could then choose to process or not process transactions through that user's pathways based on this reputation. Once identity characteristics are evaluated on the blockchain, such scenarios, in which contracts evaluate a user's viability for interaction, are near infinite.

Lastly, because UCACs may restrict which users stake their CP Tokens to a particular contract, it is possible for CP Token holders to create virtual monopolies for that contract. To illustrate, a company may build a DApp that employs the Credit Protocol, but only allow their own organization to stake CP Tokens to that DApp. Therefore, the company could charge any price they want for transactions within that DApp without worrying about price competition from other CP stakeholders.

These are merely some of the possible actions and strategies that CP token holders can pursue as they use the tokens.

Transaction Capacity Setting

Each CP Token will allow its owner to process a specified amount of transactions per day. The amount of transactions allowed per CP Token will be dynamically adjusted so that total network transactions account for 85% of the network's capacity for transactions.

This algorithm monitoring and adjusting transaction capacity may only ever increase the total number of transactions allowed per CP Token; it can never decrease it. This mechanism guarantees that available transaction capacity will grow with increasing network usage, while preventing purchasers from losing the total number of transactions allotted to them by their initial token purchase. It is akin to the expansion of broadband capacity as network usage increased with the popularity of online digital video and gaming.

1. <https://www.plasma.io/> ↔

Friend in Debt – Our First Demonstration of the Credit Protocol

With an eye toward growing the Ethereum universe, BlockMason proudly introduces a jurisdiction agnostic social platform for creating and settling debts with friends on the blockchain: **Friend in Debt** (FiD), or as we like to think about it:

“Like *Venmo* on the Blockchain, but Better”

FiD is more than an application to simplify payment between friends. As you will see, we believe in the

power of FiD to enable new types of local economies based on the liquid movement of debt and digital currency through trusted social networks. FiD provides a flexible tool for individuals, especially those without access to traditional banking, to financially engage their communities and build economic institutions that transcend political and geographic boundaries, lowering financial transaction costs, and encouraging the global development of the Ethereum ecosystem.

At BlockMason, we believe in more than just promises, which is why we invite you to test the current version of Friend in Debt **right now**. Even in this stripped-down, beta version of FiD, we hope that you will see our vision of a decentralized, social network of debt-tracking, lending, and credit extension for communities large and small. FiD enables a new and powerful system of social borrowing and microloans, one with which you and your friends can begin experimenting with **today**. We refuse to ask for your valuable time and energy without first offering a working product that demonstrates our ambition and proficiency.

Most importantly, we believe blockchain technology has enabled us to develop an essential application—one which empowers individuals to achieve decentralized financial freedom through a complex web of loans and payments within a social network.

We are building the new economy; join it.

The current version of Friend in Debt. Please [email us](#) with comments, questions, or concerns at.

More than a Wallet: Creating and Settling Social Debts

We've all been there: confronting a neighbor who constantly '*borrow*s' laundry quarters; helping a friend repay the loan for his lambo; throwing haymakers in a fistfight incited by the complications of splitting a check. Friend in Debt, at its heart, is an application for solving such problems and saving friendships. Never again must comrades argue over who paid what when, or for how much. Whether it's settling a series of dinner bills, recouping those plane tickets to Cabo, or recording invoices for official business transactions, Friend in Debt can handle any transaction you throw its way.

Additionally, boasting an interactive friends list and debt itemization mechanism, Friend in Debt combines the draw of social networking with the power of financial flexibility, a partnership that promises high user engagement, scalability, and even more efficient transactions for businesses of all sizes. Unlike applications such as Venmo, WeChat Pay, or Alipay, Friend in Debt enables a far more complex and rewarding web of transactions than simple spot settlement and immediate debt repayment.

Unlike applications such as Venmo, WeChat Pay, or Alipay, Friend in Debt enables a far more complex and rewarding web of transactions than simple spot settlement and immediate debt repayment.

Our flexibility and friend-sourced loan tracking offers a revolutionary development in the completion of financial transactions of any scale. Anyone who has ever requested a bank loan knows the pain of hand cramps or carpal tunnel caused by endless forms, the awkwardness of Inquisition-like face-to-face interrogations, and the torture of awaiting Credit Committee approval that may never come. By contrast, easier to attain microloans and payday loans are often exploitative, including high interest rates and payback schedules designed to exacerbate debt. Credit card applications, though less frustrating at the outset, too require time and energy, and no one can guarantee that efforts to secure credit lines through any of these options will result in success.

Meanwhile, each of us interacts on a daily basis with individuals we trust far more than banks, credit card companies, or payday loan hucksters. Friend in Debt makes it easy to financially engage with friends and coworkers, whether that is by tracking small social debts accrued through daily life or by managing larger loans negotiated between known parties. Before FiD, monitoring such social debt required significant mental overhead, often rendering the endeavor futile and leading to squabbles about payment history within friendships.

Existing apps that claim to alleviate such problems, such as Venmo, Wechat/Alipay, and various cryptocurrency wallets, only work for on the spot settlement—a surprisingly limited functionality when contextualized against the various complex debts and payments that can occur in a large social network. These platforms feature no way to settle debts at a later date, pay debts in increments, or automatically calculate and accrue interest, nor do they feature any functionality related to tracking and settling debts over long periods of time.

FiD will allow users to create, track, and settle debts in all major currencies on top of standard Ethereum wallet functionality, dramatically expanding the pool of possible users and use cases. In doing so, FiD effectively creates a platform for decentralized money issuance on the blockchain, only executed via a user-friendly, wallet-like interface that encourages widespread adoption.

FiD effectively creates a platform for decentralized money issuance on the blockchain, only executed via a user-friendly, wallet-like interface that encourages widespread adoption.

BlockMason launches Friend in Debt with an eye toward future development as well as present functionality. To promote swift and painless endorsement by new users, the Friend in Debt application will operate on both Desktop and Mobile, featuring hook-ins for popular app browsers including Status and Metamask. Additionally, FiD will offer an open, developer friendly backend API that permits any developer to build their own front end platform, which can be integrated into FiD. This setup enables the existence of many different front end interfaces from a variety of developers, all accessing the same back-end data. This API facilitates the smooth implementation of new features to be built on top of Friend in Debt. Therefore, if any developers conceptualize innovative methods for utilizing our platform—such as credit tracking or business reviews—they can implement their own vision via Ethereum smart contracts on top of the FiD contract base.

Unified Login Across Devices

Because Friend in Debt allows users to carry and net debts over long periods of time, with periodic or on-demand settling of debt balance through ETH, it is essential that users have the most possible flexibility in determining how to send and receive payments.

Unlike typical Ethereum DApps, FiD employs the BlockMason Foundation protocol (see [Foundation Whitepaper](#)), which unifies login and identity credentials across multiple devices and platforms (iOS, Android, and Web). Our Foundation protocol frees users from the difficulties of switching between multiple devices and platforms or passing private keys from one wallet to another, essential components of our plan to enhance user interface and increase widespread adoption.

By lowering the technical barrier to entry for users, we improve user experience and ultimately increase demand for the FiD DApp. Even more importantly, incorporating the Foundation protocol allows FiD to potentially add new users easily with a temporary Eth address—such users can later migrate to other wallets while retaining their FiD account.

Lowering the Technical Barrier

By utilizing the BlockMason Foundation protocol via the Credit Protocol, Friend in Debt addresses the most significant technical barrier to utilizing the FiD app by unifying the various wallet addresses of users under one ID. Integration with Foundation enables users to easily login to multiple accounts, wallets, and platforms without the burdensome and insecure sending of private keys. Foundation also allows for simple switching between multiple addresses whenever users choose to settle debts through Ethereum payments.

Better Security

Friend in Debt offers an improvement to existing payment applications not only through user interface and

functionality, but also through greater security. Venmo in particular has been subject to publicized exploits by scammers and problems with user accounts.

On July 19th, 2017, the Verge published a story of a Venmo scammer who stole US\$25,000 in expensive camera gear in Los Angeles: [The Verge: Venmo Scam - Fake Payment Merchant Clause Fraud](#)

Contrary to what some may believe, transactions on Venmo are not instantaneous, and payments may be canceled or rescinded even after money has been marked as delivered to another Venmo user's account. While Venmo offers various protections for users **sending** payments, Venmo offers little protection for users **receiving** funds, including those who were paid using stolen credit or debit cards, as in the above article. According to the account, Venmo first froze the victims accounts, denying them access to their existing balance. Then, Venmo informed the victims that because they had broken Venmo's user policy by exchanging goods for money, they were not entitled to any recompense.

Contrary to what some may believe, transactions on Venmo are not instantaneous, and payments may be canceled or rescinded even after money has been marked as delivered to another Venmo user's account.

Paypal, the company that owns Venmo, has been known to block users' funds in their accounts and arbitrarily kicking users off the platform. Venmo has generated further headlines by locking users out of accounts for joke descriptions of their payments. [Angelina Travels: Venmo Account Frozen](#)

With FiD, no user will have to worry about their account being locked for purposes of humor or otherwise. Because FiD uses Ethereum to settle payments, and those payments are protected by the security of the blockchain, the uncaring bureaucratic mess of large institutions bear no threat to the user—no matter how funny that user is. FiD customers will always have access to their funds, be able to settle debts with the Ethereum they own, and be able to draw from agreed upon credit lines, mitigating any risk that comes from non-cash transactions.

With FiD...payments are protected by the security of the blockchain, the uncaring bureaucratic mess of large institutions bear no threat to the user—no matter how funny that user is.

Friend In Debt Beta

The current version of Friend in Debt is already fully functional. While this iteration offers only a stripped-down, demo of the application's future capabilities, FiD beta demonstrates BlockMason's commitment to developing our DApp with vigor and efficiency. We refuse to blindly roll out a token sale, asking for ether on the strength of a Whitepaper and a promise. We have a product we believe in, and over the course of the token sale we will continue to release important updates that expand FiD's functionality.

Right now, FiD maintains a friend list and allows for the addition, confirmation, and clearance of debts. All transactions are processed on the Ropsten testnet, allowing users with access to testnet Ethereum to interact with the app completely free. The beta version runs in Chrome and requires Metamask. [fiddy.io](#)

Future Versions



Everyday, BlockMason works hard to improve Friend in Debt, and we plan on releasing several new versions in the coming year, expanding its already awesome functionality. See below a list of proposed features that we hope to add in the near future.

- Settlement of debts between wallets in ether, with autoconversion between the debt’s native currency and ether.
- Group settlement features, in which friends can join a group, add common debts, and split them amongst group members.
- The creation of payback schedules, to allow users to manage loan repayment in an easy and secure fashion.
- Autopayments for recurring debts, such as weekly gardening, children’s allowance, or alimony.
- Offchain settlement of debts to increase network speed and decrease transaction cost.

Friend in Debt – Market Opportunity:

Democratizing the Role of a Lender

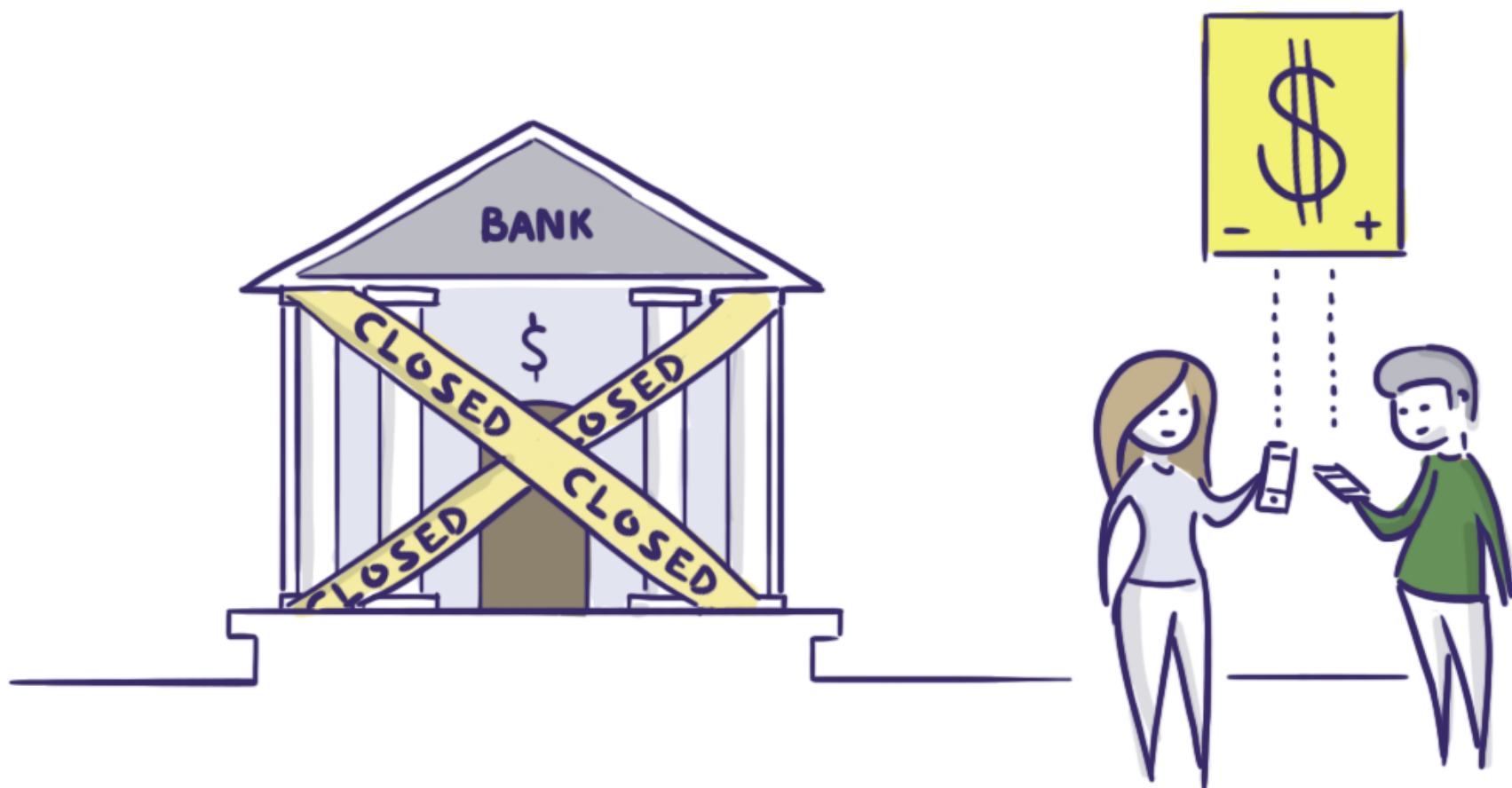
Friend in Debt is an extremely powerful tool not only for individual borrowers, but for lenders as well. As previously discussed, FiD gives each user the power to act as a lender to their friends in a jurisdiction agnostic format, allowing the unbanked not only to establish credit lines, but to individually issue credit to his or her associates—effectively acting as a self-run bank. Because FiD removes the high fees and overhead associated with traditional banking, lenders and borrowers can avoid the restrictive costs and oft-changing

associated with traditional banking, lenders and borrowers can avoid the restrictive costs and oft-changing regulations that are prohibitive of natural, free-flowing transactions that occur within a social network.

Banking the Unbanked

In wealthier economies, banks are such an entrenched institution that many are surprised to learn vast swaths of the world operate with limited or no access to banking or credit. In Southeast Asia, Africa, the Middle East, and Latin America, such financial disadvantages burden regional economies, restrict fiscal growth, and obstruct individual actors from exercising power over their financial transactions.

Despite relatively little access to large financial institutions, many in these regions possess application capable mobile phones enabled with both data and wifi. Such individuals present an obvious market opportunity for FiD, as they have a clear incentive to manage their finances in a secure, digital fashion from their phone—further demonstrated by the rise of ‘mobile phone minute currency’ in eastern Africa, a system far inferior to FiD in terms of scale and efficiency.



Even in countries with widespread access to banking, many institutions do not offer the same easy ACH transfers available in the United States. Venmo, which requires access to a bank account, has gained popularity in parts of the U.S., but cannot be used anywhere outside of the U.S. Similarly, while Wechat Pay and Alipay have grown near ubiquitous in China, non-Chinese usage of these systems is almost nonexistent.

Existing financial transaction systems across the world generally require users to pay fees for transfers. The few mobile apps that aim to improve financial liquidity require bank accounts and have limited market reach, implying a huge demand for a cryptocurrency based financial transaction application like Friend in Debt. While FiD is **much more** than a cryptocurrency wallet, even popular crypto wallets are often too complicated to promote adoption by the average non-technical user. FiD boasts a user-friendly interface and simple account creation to sign up new users with enthusiasm and ease.

Friend in Debt Technical: Credit Protocol Smart Contracts

The Credit Protocol smart contracts form the base contracts on which FiD runs. FiD is more or less an unrestricted UCAC running on top of the CP. Ergo, all of the above descriptions of the mechanisms behind the CP, apply to FiD.

Foundation forms the core of user interaction with FiD. This means that in order to use FiD you will need to

have a FoundationID. These are easy to create through the Foundation Manager UI and currently free.

Friend in Debt Use Case Authority Contract

The FiD UCAC is simply a UCAC without any rules. This means that it is open to the general public and has no limits on the amounts of debt that can pass through it.

The FiD UCAC will allow any CP Token Holder to stake their tokens and bid for hosting user transactions, to be paid in ETH. The FiD UCAC will automatically choose the lowest-priced transactions for its user, ensuring that costs are as low as possible. Therefore, users of FiD must pay a transaction price in ETH unless there are free transactions available. While owning ETH is a necessity for using current versions of FiD, in future releases BlockMason plans to develop methods for onboarding new users who don't possess ETH wallets, significantly growing the potential customer base.

Market Handler

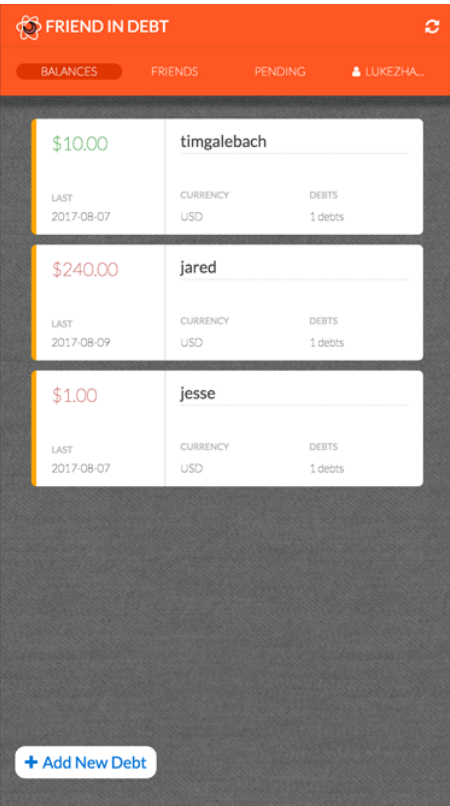
The Friend in Debt UCAC implements a smart bidding system, allowing CP Token stakers to bid for supporting FiD user transactions. In practice, CP token stakers will bid a specific amount of ETH that they'd like to charge for the transaction. The lowest bid (cheapest for FiD users) will win, if that bid has transaction capacity remaining.

Initially Blockmason will be staking a portion of tokens set aside for FiD transactions, making those transactions free. In the future, BlockMason will evaluate whether to continue this practice, use fewer CPT tokens, or raise the price of these transactions.

Front End

The front end for Friend In Debt is written using Purescript. Purescript is a strongly, statically typed language with expressive types. Purescript compiles to javascript. Purescript is very similar to Haskell. We like Purescript because of its advanced type system lets us prototype and iterate much faster. These qualities ensure writing code in Purescript is ultimately easier, more fun, and significantly less prone to bugs.

The front end allows for users to easily interact with smart contract functionality without requiring the technical expertise of being able to use raw smart contract functions. It also serves to display data in an easy to view user friendly manner.



For more information on the front end, why not try it out? Friend in Debt beta is already available for a [Test Drive](#) .

Token Sale

The Credit Protocol public token sale will last for 30 days with a goal to sell 27,000 ETH worth of tokens. The total Credit Protocol Token (CPT) supply will be fixed at 116,158,667 CPTs and no more tokens will be created thereafter. During the public sale, BlockMason will make available 33,695,200 CPTs for sale at 1 ETH for 1000 CPT, with up to an additional 50% token bonus based on timing of purchase. Bonus tokens rewarded will be counted towards the hard cap. Any uncommitted tokens will be held in reserve for future offerings.

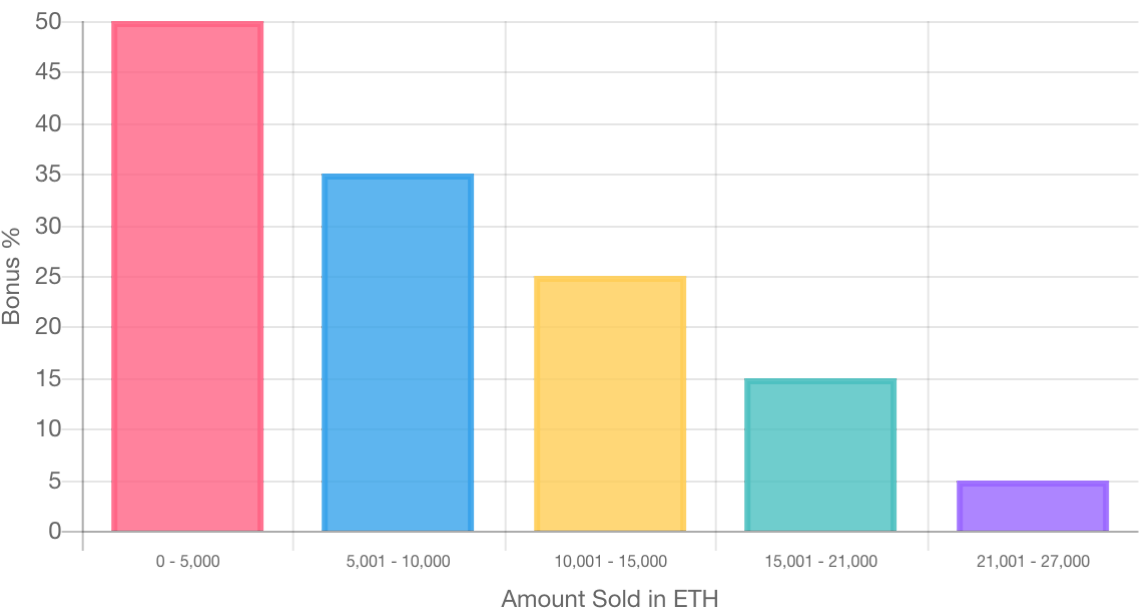
Whitelist Period

BlockMason is fully aware of issues present in many token sales, in which smaller purchasers fail to receive an opportunity to purchase tokens during the sale. Therefore, BlockMason offers to guarantee allocation of tokens to early adopters of CP, regardless of size, who purchase tokens during the whitelist period. This window serves as an opportunity for all potential customers to reserve token purchase slots, unlike other sales where “whales” and scalpers monopolize large blocks of allocation, squeezing out smaller purchases, and profiting from flipping tokens following the sale. Using a whitelist, BlockMason will record the Ethereum addresses of all users who request a guaranteed slot. Each address that requested allocation is guaranteed a chance to purchase CPT during the early 5 day period in a process executed via the smart contract governing the token sale.

BlockMason is fully aware of issues present in many sales, in which smaller purchasers fail to receive an opportunity to purchase tokens during the sale. BlockMason offers to guarantee allocation of tokens to early adopters of CP, regardless of size, who purchase tokens during the whitelist period.

Sale Bonus Tiers & Reward Mechanism

All purchases made during the whitelist and sale period will immediately count towards the token sale bonus pool count (see the chart below), and proceed chronologically in the order which they were received. Purchases made during the token sale will receive bonuses based on the sequence in which they were made, and continue sequentially. For example, if there are 12,000 ETH worth of token purchases made, the next purchaser will receive a 25% purchase bonus.



Purchase Range (ETH)		Bonus	Tokens
Lower	Upper		per ETH
Purchase Range (ETH)		Bonus	Tokens
0	5,000	50%	1,500
5,001	10,000	35%	1,350

10,001	15,000	25%	1,250
15,001	21,000	15%	1,150
21,001	27,000	5%	1,050

Developers’ Token Pool

A Developers’ Token Pool equal to 20% of the total token supply will be set aside for the BlockMason team. These CPTs will be used to seed UCACs including Friend in Debt for a minimum of 12 months in order to help the credit protocol ecosystem develop.

Friend in Debt

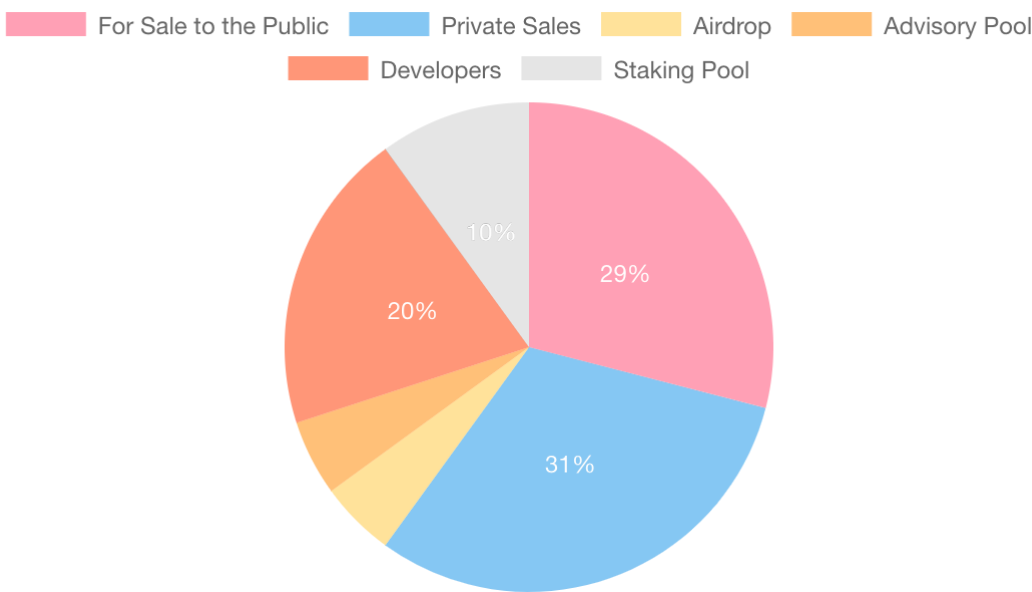
10% of CP Tokens will be set aside and used to support the transactions of Friend in Debt and any other demo DApps that BlockMason develops with the CP.

Advisors and Strategic Partners

For the purpose of providing feedback, development or marketing suggestions and services, or other services useful for BlockMason’s development, marketing and promotion of CP and FiD and for purposes of pioneering experimentation and familiarization with the CP and FiD dapps and thus in the utilizing of transactional capacity on the CP and Ethereum networks, 5% of CP tokens will be given to advisors and strategic partners.

Airdrop

In order to encourage UCAC usage and development by the Ethereum community, 5% of CPTs will be made available to all FoundationIDs on the mainnet. The airdrop will occur approximately 2 months after Foundation goes live on the Ethereum mainnet.



Use of Funds & Roadmap

BlockMason has developed a specific, concrete plan for the use and allocation of revenue from the token sale. Depending on the total amount of CPT purchased, we will execute our development plan in the manner described below. Our support and maintenance plan includes not only important documentation and customer support, but also porting of FiD to other operating systems and UCAC development for customers use in various implementations of the CP. We will continue to update and advance FiD as an open source example of how to implement the CP. We would like to reiterate that BlockMason has already released a

working **beta** version of CP & FiD, which we invite everyone to test while considering their participation in our token sale.

- 5,000 ETH of CPT Sold - Working full version Chrome Metamask FiD app and working Mobile Web App. 1 UCAC smart contract will be developed in house and released free to CPT customers. CP support docs, deployment guides, UCAC construction examples and deployment instructions available in proportion to the number of CPT sold.
- 10,000 ETH of CPT Sold - Working full version Chrome Metamask FiD app and working Mobile Web App. General purpose settlement of debts written into the Credit Protocol. All major cryptocurrencies debt settlement for FiD. 3 UCAC smart contracts will be developed in house and released free to CPT customers. CP support docs, deployment guides, UCAC construction examples and deployment instructions available in proportion to the number of CPT sold.
- 20,000 ETH of CPT Sold - Working Chrome, Mobile Web, Android and IOS FiD apps. General purpose settlement of debts written into the Credit Protocol. All major cryptocurrencies debt settlement for FiD. 5 UCAC smart contracts will be developed in house and released free to CPT customers. CP support docs, deployment guides, UCAC construction examples, and deployment instructions—scaled in proportion to the number of CPT sold.
- 30,000 ETH of CPT Sold - Working Chrome, Mobile Web, Android, IOS, OSX, Windows, Linux FiD apps with full normal Ethereum wallet functionality. General purpose settlement of debts written into the Credit Protocol. All major cryptocurrencies debt settlement for FiD. 7 UCAC smart contracts will be developed in house and released free to CPT customers. CP support docs, deployment guides, UCAC construction examples, and deployment instructions—scaled in proportion to the number of CPT sold.
- 40,000 ETH of CPT Sold - Working Chrome, Mobile Web, Android, IOS, OSX, Windows, Linux FiD apps with full normal Ethereum wallet functionality. General purpose settlement of debts written into the Credit Protocol. All major cryptocurrencies debt settlement for FiD. Interest bearing loan functionality UCAC written. 10 UCAC smart contracts will be developed in house and released free to the general public. CP support docs, deployment guides, UCAC construction examples, and deployment instructions—scaled in proportion to the number of CPT sold.
- 45,000 ETH of CPT Sold - Working Chrome, Mobile Web, Android, IOS, OSX, Windows, Linux FiD apps with full normal Ethereum wallet functionality. General purpose settlement of debts written into the Credit Protocol. All major cryptocurrencies debt settlement for FiD. Interest bearing loan functionality UCAC written. Off-chain/Plasma implementation for massive reduction in usage cost for CP. 15 UCAC smart contracts will be developed in house and released free to the general public. CP support docs, deployment guides, UCAC construction examples, and deployment instructions—scaled in proportion to the number of CPT sold.

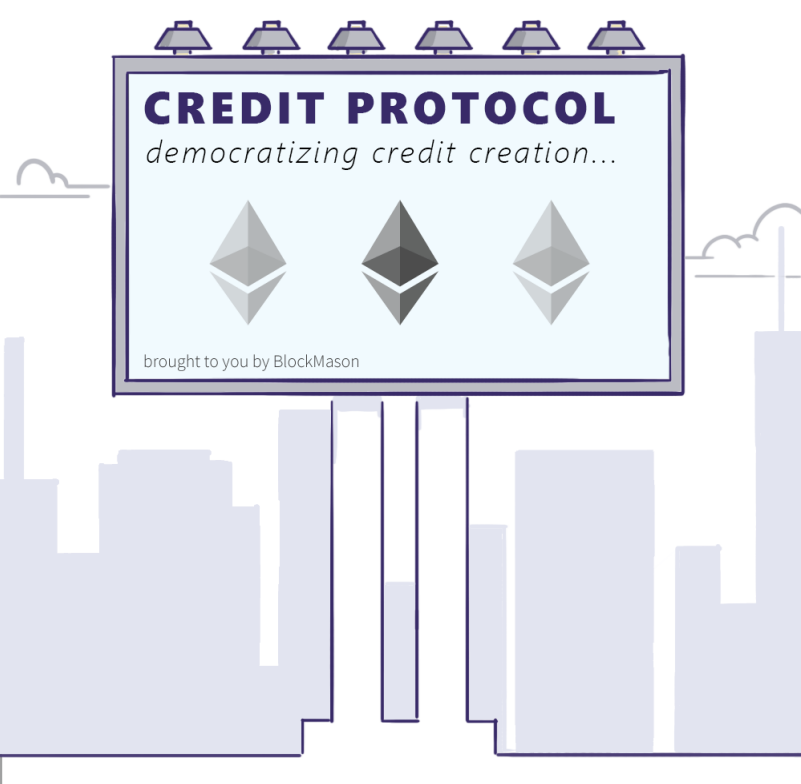
The following list outlines the allocation of proceeds towards specific uses. The upper and lower ranges of the proceeds allocation percentages reflect that, the more tokens sold, the more proceeds we will allocate towards marketing and customer support, since of course that means we will have more customers to support!.

- Platform & Application Development (35-50%): Includes software development, salaries for a development team with a goal of launching Friend In Debt on number of platforms within 6 months of the token sale close;
- Admin, Legal and Operating Expenses (5-30%): Includes salaries, office space, stationery, furniture, hardware, and legal, accounting and related professional fees;
- Marketing and Customer Acquisition (5-20%): Includes PPC, display ads, organic SEO, content marketing;
- Customer Support (10-20%): Includes documentation, smart contract maintenance, examples, demo apps, front-end hook-in instruction, email and chat support; and
- Bounty Pool (5%): A security bug bounty program similar to those run by leading tech companies. In order to find critical security bugs before they adversely affect our customer base. Includes funds for UCAC developer bounties.

Marketing Plan

While we would like to believe that the Ethereum community is an equitable meritocracy, in which the most powerful, useful applications soar to success, we know that developing software is only part of building a successful product. Application marketing is an essential task for any company, if not even more important than the product itself. Cofounders Timothy Galebach and Jared Bowie are both vastly experienced experts on tech marketing and community engagement, having worked within the industry for numerous years. Their in-depth knowledge from running multimillion dollar pay-per-click app installation campaigns influences our plans to transform Friend in Debt from a random blockchain application to a program that achieves mainstream popularity and lives atop app store popularity lists.

Both Tim and Jared have developed their own proprietary software for advertisement tracking and optimization. Even more importantly, through their many years working in the online advertisement space, they have developed an extensive network of professional connections to facilitate the most efficient marketing campaign. This may include large-scale ad buys, affiliates to encourage application installation, or contracted freelancers for banner and landing page designs.



CP and FiD Roadmap

Date	Feature	Description
2017 Q2	INTERNAL ALPHA VERSION DEPLOYED TO MAINNET	Internal Proof of Concept Beta Deployed to Both Ropsten and Mainnet With Metamask Support.
2017 Q3	COMPLETE RE-WRITE FOR V1.0.0	Alpha Version 1.0 of Chrome Metamask FiD Released on Ropsten for Private Testing
2017 Q3	RELEASE CANDIDATE	Release Candidate Version 1.0 of Chrome Metamask FiD Released on Ropsten for Public Testing
2017 Q3	ANNOUNCES Token Sale	Sale of CPT, which powers FiD .
2017 Q4	MAINNET DEPLOYMENT, VERSION 2 GOES INTO TESTING.	Contracts deployed on MainNet. Next generation contracts which supports Settlements will be deployed to TestNet.
2017 Q4	FID DEPLOYED FOR MOBILE APP BROWSER	FiD fully tested and operational in at least one mobile app browser.
2017 Q4	BASIC DEVELOPER DOCS AND UCAC AVAILABLE	Basic documentation on how to use CP . Another fully operational UCAC , offering different use case for CP .
2018 Q1	GENERAL PURPOSE SETTLEMENTS GOES LIVE.	General Purpose Settlements will be deployed to MainNet.

Date	Feature	Description
2018 Q1	DESKTOP APPS GOES INTO BETA, MOBILE APPS RELEASED.	Desktop Apps Betas released for public testing, Android and iOS Apps in respective stores.
2018 Q1	IMPROVED DEVELOPER DOCS AND MORE UCACS AVAILABLE	Detailed documentation on using CP including UCAC examples, deployment guide and basic solidity guide. 5 fully functional UCACs available.
2018 Q2	VERSION 3 GOES INTO ALPHA, DESKTOP APPS RELEASED.	Desktop Apps Released. Version 3 goes into Alpha. Interest Bearing Loans Smart Contracts released on TestNet.
2018 Q2	VERSION 3 GOES INTO BETA, LOAN CONTRACTS DEPLOY TO MAINNET.	Version 3 released to the public for Beta Testing. Interest Bearing Loan contracts deployed to MainNet.
2018 Q2	IMPROVED DEVELOPER DOCS AND MORE UCACS AVAILABLE	Detailed documentation on using CP including UCAC examples, deployment guide and basic solidity guide. 7 fully functional UCACs. available.
2018 Q3	INTEREST BEARING LOANS WRITTEN INTO FID	Interest bearing loan function written into FiD .
2018 Q3	OFF-CHAIN STATE CHANNELS GOES INTO ALPHA	Our in-house transaction aggregation and caching layer goes into Alpha Testing.
2018 Q3	IMPROVED DEVELOPER DOCS AND MORE UCACS AVAILABLE	Detailed documentation on using CP including UCAC examples, deployment guide and basic solidity guide. 10 fully functional UCACs.
2018 Q4	VERSION 3 RELEASED	Version 3 released to the public. This includes interest bearing loans and massively reduced transaction costs thanks to off-chain state channels.
2018 Q4	IMPROVED DEVELOPER DOCS AND MORE UCACS AVAILABLE	Detailed documentation on using CP including UCAC examples, deployment guide and basic solidity guide. 15 fully functional UCACs.

Security (General)

Writing code for Ethereum is a new and complex endeavor. Because of its novelty, and the general difficulty of keeping untested code 100% secure, the Ethereum community has recently witnessed multiple large scale hacks. BlockMason is fully committed to avoiding the mistakes of these compromised endeavors, and we have taken the following steps to keep our application and token sale secure:

- Write secure code according to best practices.
- Slowly rollout all new features and improvements, first locally, before proceeding to the testnet, and finally to the mainnet.
- Professionally audit any code interacting with Ethereum or [CP](#) tokens.
- Offer a ‘bug bounty’ to encourage whitehat hackers to identify and eradicate coding errors.
- Employ [ENS](#) addresses, make available token sale contracts prior to the sale, and establish clear and secure communication channels.

These steps comprise only the beginnings of our final security plan. BlockMason is committed to constantly researching and evaluating threats faced by other DApp companies, and to adjust our security measures accordingly. We have invested a tremendous amount of time, energy (and *love*) into CP and FiD, and we would be personally devastated at the emergence of any serious security issue. Our reputation, and your trust, is on the line—therefore, we approach issues of security with the maximum level of consideration and caution.

FAQ

All transactions are visible on the blockchain. How can I keep my transaction history private?

While transactions on the blockchain are currently public, the Ethereum Foundation and the team behind ZCash, an anonymous cryptocurrency, have partnered to develop Project Alchemy, an undertaking aimed to providing anonymous blockchain transactions. Eventually, transactions will only be visible to specific users, to whom the sender allows visibility. With this protocol enabled, all problems of public and visible transactions disappear. In the current version of Ethereum, *Homestead*, all transactions will be visible, and users should only post what they are comfortable sharing.

*Isn't it **Expensive** to Send Transactions on the Blockchain?*

When reserved for sending only Ethereum currency, transaction costs on the Ethereum network are extremely low, especially when compared to other options such as Bitcoin, bank transfers, or credit cards. On the other hand, storing data about debts, lenders and borrowers, and currency information is significantly more expensive.

One of our immediate goals is to lower this cost. In our ideal version of CP, users need not ever consider transaction costs. Even better if users don't need to know the intricacies of Ethereum and gas pricing. The system will be as low cost as possible in order to allow for easy adoption by non-technical users.

We will employ either a Raiden Network like or our own in-house off-chain solution to significantly lower transaction costs. Such off-chain solutions can reduce costs per transaction over one hundred times.

Isn't the Ethereum Network Slow?

One of the Ethereum Foundation's major goals is to is to massively increase network speed and scalability. In the future, it is not unlikely that Ethereum will easily handle tens of thousands, or even millions, of transactions per second directly on-chain or off through networks like Raiden. In the meantime, Ethereum's future POS system has been reported to be mining blocks at a rate of one per three seconds. Raiden, and other forthcoming off-chain solutions, promise much faster speeds. Between network scaling, Plasma, and the development of off-chain transaction options, the Ethereum Network will likely see a dramatic increase in transaction speed, a critical step for any DApp with a large user base.

What Happens on CP and FiD If a User's Address is Hacked?

Unfortunately, hacking is a reality of the current underdeveloped Ethereum ecosystem. We are fully conscious of the threats facing Ethereum users, and have spent a great deal of time considering the security of Foundation users. To that end:

- CP and FiD will never hold any amount of Ethereum. Therefore, hacking a CP or FiD user account offers

no Ethereum reward. Ethereum is stored in a user's wallet or address as normal.

- Debts await confirmation from both sender and receiver before a transaction is finalized. A hacked account cannot simply send off debts and hope for a payout.
- Settlement of debts is a manual process, which starts from a single address. This address and it's Ethereum are secure and separate from any Foundation account. It is a normal Ethereum address.

The Foundation protocol smart contracts separate data from logic. As we audit the security, we can update Foundation without losing a user's underlying data. Foundation is a necessary component of the CP, and we look forward to community auditing of the open source code to continue improving its security.

What Happens with Remaining Tokens Not Purchased Through the Token Offering?

Remaining tokens not sold during the first token offering will be held by Blockmason and sold at a future token offering. Blockmason intends to hold 20% of the CP Token supply and use an additional 10% for staking FiD and other demos developed by BlockMason using the CP.

What Happens With Friends Who Don't Pay Their Debts?

Friend in Debt doesn't offer protection for consensual financial transactions. For untrustworthy friends, we can only recommend you either immediately settle debts or don't extend them credit lines.

We won't arbitrate your friendships—we only have so many tough guys with baseball bats, and they can't be everywhere at once.

Team Profile

Timothy Galebach (Co-Founder)

Timothy is a startup founder, technical expert, and business development specialist, with a particular focus on data-driven ventures and automated infrastructure. He is currently the owner and founder of Almond Media Solutions, a technology consulting firm in the e-commerce and display advertising space.

Timothy graduated from Harvard University in 2007, with a degree in Computer Science, and went on to work with a variety of startups and established companies in CTO roles. His primary focus was making large amounts of data visualizable and actionable for salespeople and internal business teams, and eliminating cost centers through aggressive optimization and automated data orchestration.

Jared Bowie (Co-Founder)

Jared Bowie has been designing systems to coordinate distributed databases and peer to peer networks for more than 10 years. He was an early adopter of Bitcoin and invested in multiple altcoin mining operations. He became fascinated with the power of Ethereum when he began writing programs on its blockchain and is now convinced of its power to take over the world.

Jared graduated from James Madison University with a major in philosophy. After graduation, obsessed with the convergence of big data and display advertising, he started his first digital advertising venture. Since then, Jared has written specialized software and proprietary algorithms to run successful multimillion dollar ad campaigns and manage remote teams based in Hong Kong, mainland China and India.

Michael Chin (Co-Founder)

Michael graduated from Washington University in St. Louis magna cum laude with majors in Finance and Accounting. He has spent his entire career in the credit industry, beginning with J.P. Morgan’s principal mezzanine credit group in New York. Michael then transferred to JPM’s Debt Capital Markets and Syndicated Leveraged Finance group in Hong Kong, helping Asian issuers with Investment Grade, High Yield debt, and syndicated loan offerings. Then, Michael joined the Hong Kong office of Intermediate Capital Group (a FTSE 250 composite company), one of the pioneers of mezzanine debt in Europe, before helping co-found BlockMason.

Luke Zhang (Lead Developer)

Luke is an experienced developer and designer, whose previous accomplishments include building an automated advertising tool that handles millions of dollars in spending. He also co-programmed a globe-spanning system to track product components across the supply chain, processing over \$400 billion annually. Educated in Canada and China, Luke has been an avid user and fan of Blockchain technology since 2011, and he has enjoyed the rollercoaster ride of the technology’s evolution ever since. Luke recently dedicated his life to fully developing applications on top of the Ethereum network, under penalty of death.

Consultants

Jesse Nee-Vogelman (Content Master)

Jesse, the voice of his generation, graduated from Harvard University in 2013 with a degree in Slavic Languages and Literatures. He has spent the last two years as the Artist in Residence of Harvard’s Signet Society of Arts and Letters, completing a novel. Previously, he worked as a freelance writer and writing consultant in South Korea and the United States. His fiction has been published in The New Haven Review and The Harvard Advocate, for which he won the Louis Begley Prize. His music reviews and features have appeared on Pretty Much Amazing, Slant Magazine, Spectrum Culture, and The Daily Distiller. He enjoys both cats and dogs.

Teddy Ninh (Original Art)

Teddy Ninh is a young artist with a gleam in his eyes from the moisturizing eye drops he always uses. He is currently studying at Harvard College, where he is the Art Director for the Harvard Lampoon, where he gets mediocre grades, and where he is barely a student.

Ola Vikholt (Backend Developer)

Ola is a master of his craft—an expert in electrical, software, and music engineering. After studying at the University of Tokyo, Ola completed a Masters in Signal Processing and Communication at the Norwegian University of Science and Technology. After completing his thesis, Ola worked as a software engineer for Google Japan, and has continued pushing the boundaries of technology as an engineer and project manager for several high-level tech companies in Japan.

Ola is an experienced coding teacher and computer language savant. His only purpose in life is to make the BlockMason backend systems bulletproof.

Steve Cheung (Backend Developer)

Steve graduated with honors from the world famous Computer Science department at the University of Waterloo, which has produced many of today’s leading minds in the field of artificial intelligence. Since then, Steve has headed dozens of projects as a lead software developer for cutting edge startups across the world, and he specializes in designing, developing, and deploying web applications for his clients with Ruby on Rails. In his free time, Steve has contributed extensively to the world of open source software, expanding access to quality applications and programming tools.

Steve recently declared his only passion is to optimize the BlockMason server infrastructure and build our

we're recently decided the only path forward is to spin out the BlockMason server infrastructure and sell our backend servers.

Conclusion

Unlike the products of many token sales, the Credit Protocol offers more than a promise — it is a functional protocol that provides immediate value, while offering a concrete path to widespread user adoption and significant benefits for token purchasers. It has an obvious market and clear use cases, with little to no barrier to entry for potential developers and their user base. It fills a glaring hole in the Ethereum ecosystem and provides a platform for future developers to implement their own revolutionary ideas. The Credit Protocol is simultaneously bold and secure, audacious and deliberate, and, most importantly, it works right now.

Even more exciting, the Credit Protocol and Friend In Debt represent an objective step forward toward our vision of a world run on Ethereum. Debt is a vital factor in the growth of any economy, and without moving debt to the blockchain, the cryptocurrency ecosystem has no choice but to stagnate. The Credit Protocol enables the next phase of Ethereum's development, one in which the creation of large-scale projects and infrastructure becomes inseparable from the blockchain.

Implementations of the CP can democratize currency creation by recording mediums of exchange, units of account, store of value, and standards of deferred payment. In fact, because an individual need not own ether in order to draw debt in ether, it is entirely possible to imagine a future in which debts and credits recorded on the Credit Protocol could exceed the total market capitalization of all cryptocurrencies combined - dramatically increasing the scale of the digital economy.

Similarly, Friend in Debt is an application built for the present and designed for the future. It promises a global network of decentralized transactions that empower individuals left behind by the bureaucratic, profiteering mess of large institutions like banks and Paypal. It trusts in the ability of the individual and the force of the multitude, empowering single actors to seize their financial destinies while encouraging economic development in communities ranging from families in Minnesota to villages in Mauritania.

We at BlockMason believe in the future of Ethereum, but Ethereum's future needs the Credit Protocol—a platform that will draw in users across the globe previously uninitiated in Ethereum, and realize the potential of Ethereum not just to support a popular cryptocurrency, but to engender an entire ecosystem of apps. And when Ethereum wins, we all win.

Advisories

Purchasers of BlockMason's CP product-use tokens receive a perpetual, fully-paid-up license to use a portion of CP network capacity, proportionate to the number of tokens purchased, pursuant to an industry-standard licensing agreement. Title and ownership of Foundation, CP, FiD and all related documentation and support structure resides in BlockMason. BlockMason reserves the right to adjust the amount of network capacity, and the number of transactions allocated to each CP token, using reasonable efforts to maintain efficient CP network operation and avoid over-capacity. Our sale of product-use tokens is not a solicitation of investments. Purchases of our product-use tokens should be based on each purchaser's evaluation of our products and each purchaser's intended use and development of those products. Our products are offered "AS IS" - purchasers make their own judgments based on the beta versions and documentation for the products. We make no warranties, and we expressly disclaim all warranties, express or implied, including the warranties of merchantability, fitness for a particular purpose, and noninfringement. We cannot guarantee against service interruptions or make warranties about the Ethereum network. Purchasers are responsible for consulting their own country's laws and regulations.

Our legal counsel has issued an opinion letter concluding that U.S. securities law presents no reason to prevent persons in the United States from participating in our token sale.

Our Full Legal Opinion

