Why DLCs are a poor fit for Bitcoin-backed lending

The Firefish Team

Abstract

Discreet Log Contracts (DLCs) are often presented as a natural building block for non-custodial Bitcoin-backed lending. In this paper, we challenge this view and argue that DLCs are a significantly weaker fit for Bitcoin-backed lending than is commonly assumed.

Introduction

An increasingly popular use case of Bitcoin is Bitcoin-backed lending. In a standard Bitcoin-backed loan, the borrower locks Bitcoin as collateral and receives fiat or stablecoins from a lender. The main idea of a Bitcoin-backed loan is that the value of the Bitcoin collateral must be higher than the loan amount throughout the lifetime of the loan. This guarantees the lender that they will always get paid: the borrower either repays (and the collateral is returned to the borrower) or the borrower does not repay, in which case the Bitcoin collateral is sold to cover the debt (called default).

For that to work, there must be a mechanism called price liquidation. Should the Bitcoin price decrease so that the value of the collateral stops covering the loan, the Bitcoin collateral must be sold (liquidated) to cover the debt, similarly as in the case of default. The Bitcoin price at which the price liquidation happens is called a liquidation price. The entity responsible for triggering the price liquidation is usually called a price oracle.

On the other hand, the borrower can add more collateral (called *collateral top-up*) to lower the liquidation price and avoid liquidation.

Custodial vs. non-custodial loans

Bitcoin-backed loans can be broadly divided into *custodial* and *non-custodial* designs. In custodial lending, the borrower transfers Bitcoin to a custodian who has full control over the collateral for the lifetime of the loan. Typically, the custodian acts as both the lender and the price oracle. The custodial approach has its undeniable advantages: loan setup is simple, and the liquidation process

is straightforward. Its central drawback, however, is full counterparty risk. For many borrowers, it outweighs all the advantages.

In non-custodial lending, the goal is to avoid unilateral control of the Bitcoin collateral by any single entity. That is usually achieved by separating the role of the lender and the price oracle. Most non-custodial designs make use of some kind of multisig, typically involving the borrower and the price oracle (or/and the lender).

Discreet Log Contracts (DLCs)

Discreet Log Contracts (DLCs) are a technique that allows Bitcoin payments to be conditioned on real-world data. In our previous paper [1], we explained how DLCs work in detail. Here, we would like to recall the main properties of DLCs:

- **Scalability:** The oracle does not interact with any contract. It only attests to the outcome of an event and publishes the attestation in some public feed.
- **Privacy:** The oracle does not know any details about contracts conditioned on it. In fact, the oracle does not even know that the contract exists at all.
- **Accountability:** Any number of contracts can be conditioned on one event. The oracle cannot attest to two different outcomes of one event without compromising its private key.

In the Bitcoin community, DLCs are often presented as a natural building block for non-custodial lending, improving the standard multisig setup. At first glance, DLCs may appear to be a natural fit for this use case: price liquidation can be viewed as a Bitcoin payment conditioned on real-world data, where the relevant real-world data is the Bitcoin price.

In this paper, we challenge this widespread view that DLCs are a natural building block for non-custodial lending. We focus on limitations of DLCs and argue that the match between DLCs and Bitcoin-backed lending is weaker than it initially appears.

Limitations of current DLC designs

Price liquidation

The current specification and implementation of adaptor-based DLCs require that an event be announced ahead of time, including its public key (nonce), date, and possible outcomes. Such a requirement is reasonable and makes perfect sense for use cases based on events that will eventually happen. Classical examples of these use cases include sports betting (e.g., a tennis match occurs

on a specific date and someone has to win) or futures contracts (the underlying asset will have some price at the settlement).

On the other hand, there are real-world use cases using events that (i) do not have a specific date on which they should happen and (ii) it is even uncertain whether they will happen at all. Price liquidation in a Bitcoin-backed loan is a natural example. If the value of the Bitcoin collateral stops covering the loan due to the decrease in the Bitcoin price, the Bitcoin collateral must be liquidated. **But liquidation is not tied to any fixed date: it may occur at any point during the life of the loan, or it may never occur at all.**

Example. It could be possible, to some extent, to cover the liquidation event using oracles based on the current specification. For example, consider an oracle that attests to the daily Bitcoin price—specifically the lowest value of the day—with a 1 USD granularity. Suppose there is a one-year loan with a liquidation price of 60,000 USD. Then, the borrower and lender would need to construct one Contract Execution Transaction (CET) for each possible liquidation, i.e., each combination of 'date' (365 possibilities) and 'price' (60000 possibilities), resulting in approximately $365 \times 60000 \approx 22$ million CETs, which renders the construction operationally impractical.[1] Moreover, this construction would introduce an additional drawback: in the worst case, liquidation could be delayed by up to 24 hours.

Accordingly, we do not see a way to support price liquidation with a reasonable number of CETs while preserving the *scalability* property of DLCs.

Collateral top-up

A second problem appears as soon as one focuses on collateral top-ups. Collateral top-up is a core mechanism of a Bitcoin-backed loan. It allows the borrower to add more Bitcoin as collateral to avoid price liquidation. A collateral top-up lowers the liquidation price.

The problem is that the original liquidation price is already hardcoded in CETs via the anticipation points. As the oracle is not aware of contracts based on it (*privacy* property), it is not possible to stop an oracle from attesting to the original liquidation price should that price actually be reached. Therefore, in the case of collateral top-up, Bitcoin collateral must be moved into a new UTXO to avoid being liquidated at an outdated liquidation price.

The same issue arises in the case of partial repayment, although partial repayment is not fundamental to Bitcoin-backed lending in the same way that collateral top-up is.

Accordingly, we do not see a way to handle collateral top-ups without closing the original contract while preserving the *privacy* property of DLCs.

[1] The 60000 possibilities might be reduced by numerical optimizations defined in DLC specification down to $\log_2(60000) \approx 16$, but even the optimized variant remains rather impractical and still possesses the other disadvantages described below.

Oracle infrastructure is not ready

Oracle dependence is a core constraint

Price oracles play a crucial role in the Bitcoin-backed lending ecosystem, ensuring that the value of the Bitcoin collateral is always high enough to cover the loan. If a price oracle ceases to operate, price liquidation may no longer be enforceable, exposing lenders to significant risk. If a price oracle is compromised, it may trigger liquidation at an incorrect price, exposing both borrowers and lenders to risk. That emphasizes the importance of price oracles for Bitcoin-backed loans.

Moreover, in DLCs, **once a loan is set up, the oracle is fixed and cannot be changed.** We recall that this is because the Contract Execution Transactions (CETs) are pre-signed using that specific oracle's public key. Such dependency is in contrast to, for example, traditional software—if a library is found vulnerable, it can be immediately replaced by another one.

This inflexibility matters much more in Bitcoin-backed lending than in short-dated use cases. For example, a sports-betting DLC might only rely on an oracle for a few hours or days. On the other hand, a Bitcoin-backed loan may rely on an oracle for a year, two years, or even longer.

Incentives and competitiveness

There is also an economic layer that is easy to miss in purely technical discussions. Because price oracles are so important for Bitcoin-backed loans, they should have strong incentives to operate continuously and honestly. Ideally, those incentives would combine reputation, economic stake, and fee revenue.

But a lending platform also needs its own fee model. Even if oracle fees are individually modest, they stack on top of the other costs of the loan. As a result, the competitiveness of a DLC-based lending product depends not only on whether the protocol works but also on whether the combined oracle-plus-platform cost structure remains attractive relative to simpler alternatives.

Examples from industry

The case against DLCs as a natural foundation for Bitcoin-backed lending is not merely theoretical; it is also reflected in how real products have been designed in practice. There are Bitcoin-backed lending platforms that have mentioned plans to use DLCs but ultimately did not deploy them.

Lendasat: They originally planned to use DLCs as per their whitepaper [3], but this never reached production. Instead, they decided to use a 2-of-3 multisig setup.

Lava: They also originally planned to use DLCs [2], but this never reached production. Instead, they shifted toward a fully custodial approach.

Lygos: They claim to be using DLCs [4]. Nevertheless, their implementation departs materially from the standard DLC model. Concretely, in their setup, the oracle knows about each loan. That gives them enough flexibility to handle the issues with price liquidation and collateral top-ups discussed in this paper. But as a consequence, they forgo the three core properties that define DLCs: *scalability*, *privacy,* and *accountability.* Their solution therefore cannot be regarded as a pure DLC-based solution.

Conclusion

DLCs are an elegant and powerful construction for conditioning Bitcoin payments on real-world data. For simple short-term contracts such as betting, they are a natural fit.

Bitcoin-backed lending presents a materially different set of requirements. A Bitcoin-backed loan is not merely a conditional payment at a specific date. It is an ongoing collateral-management process that includes borrower top-ups and price liquidations that may occur at any time. And the entire system depends on oracle infrastructure remaining available and credible throughout the life of the loan, which may span several years.

As we have shown, the common intuition that "DLCs are obviously the natural way to build non-custodial Bitcoin-backed loans" is not supported by current DLC designs. At the same time, we believe that the principles underlying DLCs may still inform better architectures for Bitcoin-backed lending in the future and therefore remain well worth further investigation.

References

[1] Demystifying discreet log contracts (dlcs). `https://firefish.io/resources/demystifying-discreet-log-contracts-dlcs`.

[2] Lava loans protocol v2. `https://bitcoinmagazine.com/technical/lava-loans-protocol-v2-dlc-based-bitcoin-collateralized-loans`.

[3] Lendasat whitepaper. `https://github.com/lendasat/whitepaper/blob/master/lendasat-whitepaper.pdf`.

[4] Dlcs are perfect for lending. `https://mblack.io/posts/dlc-are-perfect-for-lending/`.