Financial contracts on distributed ledgers

Nick Van den Broeck, Casper Association

February 5, 2024

Contents

1 Introduction **2**

2 The importance of smart financial contracts **3**

3 Use cases for ACTUS on Casper **4**

4 ACTUS **6**

5 Why we need an $L2$ **7**

6 Zero knowledge proofs **8**

7 What is needed for a ZKR $L2$? **9**

8 MVP **10**

9 Long-term vision **11**

10 Conclusions **12**

1 Introduction

Over the last decade, the blockchain industry has been moving forward with smart contracts, allowing DeFi to program currencies adhering to "code is law". However, most real-world financial contracts don't fit within this framework. Rather, they require a jurisdiction to be associated with the contracts. Consider as an example a loan, where the loanee could decide to stop making monthly payments. In such a case, the loaner requires tools to keep the loanee accountable, escalating up to legal measures. In practice, our modern legal escalation process is so effective that this escalation is an edge case. In contrast, it is impossible to make code force the loanee to make monthly payments or to take into account legal concepts such as bankrupcy, unless overcollateralization is required.

As we can see from this example, in order to grow into a plethora of use cases, the blockchain industry requires pushing beyond smart contracts into smart financial contracts. These are on-chain records detailing financial agreements which live within a set jurisdiction. In essence, a smart financial contract is an agreement from which a set of future cashflows can be derived which includes the mediation of a specific jurisdiction.

The creation of smart financial contracts rests upon two innovations: A standard for describing financial contracts in code, and a way to extend the transaction throughput and privacy controls on a blockchain. In this essay, we will describe both of these innovations as well as how to combine them in order to reasonably implement smart financial contracts.

This essay starts by explaining why one should care about smart financial contracts in the first place. After discussing some use cases, section 4 digs into the first innovation, the ACTUS standard on financial contracts. Sections 5 and discuss Casper's need for an L2 and the innovation we require for this L2: Zero-knowledge proofs (ZKPs). We explain what a ZKP-based L2 requires and its general structure. We complete the story in sections 8 and 9 by exploring what such an $L2$ will look like for the Casper blockchain as an MVP and in the long-term vision. Finally, we will present our conclusions. Technical details on the Casper Association's implementation plans are left for a future essay.

2 The importance of smart financial contracts

So now to the crux of the introduction: Why should you care about integrating ACTUS with the Casper blockchain? There are three parts to this question: Why would the financial industry care about ACTUS, why on a blockchain, and why does this matter for Casper?

To start things off, let's discuss why the financial industry cares about ACTUS. The industry has significant issues. As James Grant put it, "Progress is cumulative in science and engineering, but cyclical in finance." This is a clear sign that there is something wrong. The system seems to go through cycles of increasing and decreasing regulation without learning the right lessons. One of the ways to break through this issue is to increase transparency in financial institutions and their risk both for the institutions themselves, for external risk analysts, and for regulators. This requires two things: Standardization of regulatory reporting for financial institutions, and for the content of the reports to be usable as a basis for analyses. Currently, financial reports mostly discuss the company's accounting situation, which provides meaningful insights but prevents many forms of analysis. Rather, we should look at the fundamentals of these institutions, i.e. their financial contracts, and use these as a basis for reporting. Secondly, a standard is required such that reporting of information on financial contracts is homogeneous. This will allow anyone to run their own risk analysis on important institutions, both internally and externally to the company, in order to keep them accountable and provide natural feedback mechanisms against excessive risk.

Why would the financial industry want the ACTUS protocol to be integrated with a blockchain? The answer is privacy, security and scalability.

@Mark: Could you write a paragraph here? (Auditable trust) I'm assuming the goal here is to allow the financial institution to expose enough information about itself so people can assure themselves they're doing well, while also protecting the privacy of their customers and things like investment strategies.

Finally, why does any of this matter to the Casper community? The Casper blockchain is very well-built, but does not have the first movers' advantage in smart contracts-based blockchains. Therefore, Casper must do something in order to set itself apart and provide a unique value proposition. The combination of an exploration of zero knowledge-based L2 technology and attracting people through the plethora of use cases in the financial industry, gives the Casper community the tools to pursue significant growth while adding clear value to one of the most important industries.

@Mark: What do you think about the last paragraph?

3 Use cases for ACTUS on Casper

Risk analysis has been performed in the financial industry for a long time. However, an important rule in software and any form of analysis, is "garbage in, garbage out". Without a large amount of high-accuracy, high-relevance data as input to the analysis, nothing meaningful can be derived. This leads us to three problems with modern-day financial reporting.

First of all, external risk analysts and financial regulators do not have access to raw data pertaining to financial institutions, but rather to the data reported by these institutions. This reporting happens mostly for accounting purposes, and its format is adapted to that goal. The data is therefore both low-relevance from a risk analysis perspective, and much more open to being influenced by the institution in order to manipulate the risk analysis. For example, Silicon Valley Bank's accounting showed no issues whatsoever until very briefly before it entirely shut down, leaving a gaping hole in the economy and reputation of the financial industry. Therefore, we need a new way of reporting information, not based on accounting principles but rather directly discussing the relevant unit of information here: The financial contract.

Secondly, there is currently no homogeneity in the storing and reporting of this high-relevance data. This makes it very difficult to run risk analyses, since there is significantly reduced opportunity for collaboration between institutions, as shown in figures 1 and 2. In the former figure we observe a situation where data is stored and communicated in a heterogenous way.

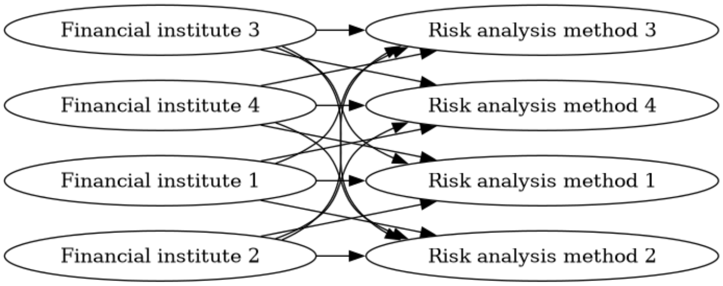

Figure 1: Non-standardized risk analysis

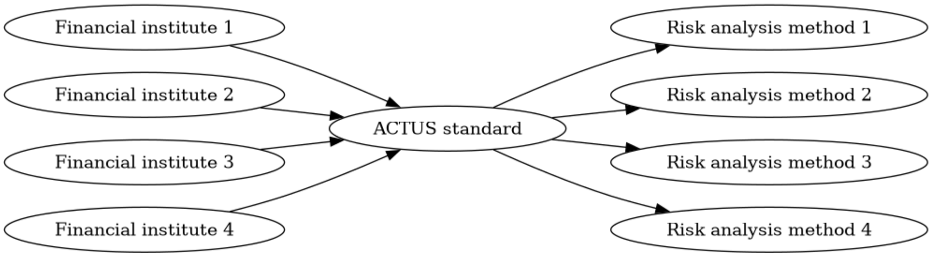

Figure 2: Standardized risk analysis

As you can see, anyone who wants to run a new form of risk analysis on N financial institutions in this situation, has to write code to convert the data of N financial institutions into a format he can use for his risk analysis. Introducing the ACTUS standard would convert this unmanageable amount of work into simply turn the ACTUS data into the input format of the new risk analysis.

@Mark Explanation of potential use cases for the system, such as for trading and settlement in decentralized exchanges or for managing financial contracts in traditional financial institutions

At Casper Association, we want to provide the tooling to help financial institutions adopt the ACTUS standand and communicate their financial information while preserving sensible privacy options. Eventually, some financial institutions could even decide to adopt Casper as their core software, as a trusted, decentralized version of a mainframe.

4 ACTUS

As mentioned above, the financial industry is in need of a standard in which to store and communicate financial information The standard must fit certain requirements:

1. Express the information compactly
2. Allow for sensible compression throughout analyses
3. Form a solid basis for analyses by both regulators and risk analysts
4. ideally, allow financial institutions to derive their accounting from the information reported in the standard

The ACTUS standard proposes that the correct unit for such a standard is the financial contract.

Why exactly should the financial contract be considered the atomic unit of the global economic system?

A financial contract is a set of terms and a set of external dependencies[1] which lead to a set of predicted cashflows between the agreeing parties. The ACTUS standard defines a taxonomy of financial contract, breaking down the choas into 32 different contract types, each with algorithmic future cashflow generation. See the ACTUS website for more information on the standard itself.

The ACTUS standard fits all of our criteria. First of all, each contract is described as a set of terms, which are numbers, and an update function which depends on external sources, which are numbers. Therefore, all the data we want to store, is the contract type and a set of numbers per contract. This forms a compact expression of the information.

Secondly, for the purpose of running a risk analysis, any contracts of the same type and with similar terms, can be approximated by one larger contract of the same type. This allows for significant compression in a world where millions or even billions of contracts could exist within a single isntitution.

[1] These are things such as stock prices.

Thirdly, ACTUS-based reporting forms a basis for risk analysis, since all the relevant financial information is present in order for an analyst to run a simulation against any scenario[2].

Finally, a firm's bookkeeping can be entirely derived from ACTUS-based reporting, since no financial information is lost.

5 Why we need an *L*2

Before we delve into the blockchain side of this project, let us recap what we have discussed so far. Firstly, the financial industry is in need of transparency. Secondly, this need can be resolved by combining the ACTUS standard on financial contracts with a purpose-built L2 on top of Casper. Now it is time for the third part of the equation: Zero-knowledge proofs as a way to build an L2 for casper.

Firstly one could ask why we need an L2 on top of Casper in order to integrate ACTUS. The main reason for this is Casper's current throughput restriction. Any financial institution logging its contracts on Casper would store millions of contracts, which Casper's L1 simply cannot handle quickly enough. In addition, this project requires specialized privacy controls not offered by Casper currently.

ACTUS on Casper thus requires an L2. There are several types of L2 solutions, including optimistic rollups, zero knowledge rollups and sidechain-based systems. We at the Casper Association decided to focus on Zero knowledge Rollups for a number of reasons:

- They seem to be the most feasible path forward.
- They allow more finegrained privacy control than most other options.
- They allow for larger scripts to be executed than with most alternatives, since the scripts themselves never touch the L1. Of course the proof size and construction time are limiting factors here.

In the next few chapters, we will create some intuition for what zero knowl-

[2] A scenario to run a risk analysis on, consists of values for all the relevant external sources, such as stock prices and interest rates.

edge proofs are and what constitutes a zero knowledge-based L2, or so-called zero knowledge rollup.

6 Zero knowledge proofs

Zero knowledge proofs (ZKPs) are an implementation of the idea that you can prove you have something without revealing the thing. A common example of this are public/private key-pairs. In this example, I can generate a key-pair and prove to you that I have the private key without revealing it. This works as follows. First I tell you which public key I have, which is associated with my private key. Next, you generate some bytestring and send it to me, and I sign it using the private key. You can then verify the signature using the public key, of which I am claiming to have the associated private key, and thereby confirm that I do have the private key. However, the signature itself cannot be used to easily discover the private key itself. This is called a zero knowledge proof, i.e. I prove to you that I have something without revealing the thing I'm proving I have.

Zero knowledge proofs have two great properties, namely compression and privacy control. Firstly, it is possible to generate proofs which are shorter than the secrets they are proving[3]. This means ZKPs can be used as a way to compress information. One of the major problems in the blockchain industry is that each (worker) node needs to store all the information on the chain. With ZKPs we can instead generate a proof that we have a valid transformation turning the blockchain from a given state into a new state, committing the proof itself to the blockchain. The full transactions, on the other hand, can either be kept secret or be posted publicly in a less consensus-requiring environment, such as IPFS.

Secondly, since zero knowledge provers can both take in public inputs and private inputs (i.e. the secrets they are proving), we create more granular control over what is kept private about the transactions we want to submit to the blockchain. This level of privacy control is necessary in many

[3] Note that this is provably impossible in the general and exact case. However, ZKPs are probabilistic, meaning that they don't bar false positives. In practice this is something to keep in mind when designing ZKP systems, to make sure this doesn't become an issue in practice.

contexts, including financial institutions logging their partly private information while revealing the information that must be reported on.

The final notion that must be discussed in this chapter, is that of Zero Knowledge Rollups (ZKRs). The idea behind a ZKR is that instead of committing a proof of a given transaction to L1 to update the blockchain's state, we can roll up a large number of suchs proofs into one proof. This rollup of zero knowledge proofs can then be committed to the L1, significantly reducing the amount of information each L1 (worker) node must store and verify.

7 What is needed for a ZKR *L2*?

In this chapter, we will discuss what is needed in order to build a ZKR-based L2 on top of Casper. Such a system consists of six components:

1. Contracts
2. ZK prover
3. Consensus layer
4. Data availability layer
5. Rollup: Compress ZKPs
6. L2 nodes

The first challenge is to turn financial contracts into code. This is solved by the ACTUS standard, as discussed in section 4. This requires us to provide a Rust implementation of the ACTUS standard, so we can more easily and securely interact with the standard from a Casper-friendly codebase.

Secondly, we need to implement circuits in a ZK prover to generate and verify proofs for ACTUS contracts. We are currently exploring different prover systems, including Halo2, Risc0, OSL, and Noir. More information on the results of this exploration will be included in a blogpost written after the exploration concludes.

The third component is the consensus layer, or Casper's L1. Building an

L2 ZKR requires the L1 (worker) nodes to recognize when a transaction comes from a L2 node, and to run the correct ZK verification process on the proof in order to validate the transaction.

Fourth, a data availability layer takes care of revealing any information about the transactions which is both allowed to be revealed and not included in the rollup posted on the L1 chain.

Rollup software must be implemented to roll up a large number of ZKPs for ACTUS-based transactions into one proof.

Finally, the L2 itself consists of one or more nodes. These nodes expose an API in order for submitting ACTUS-based transactions, combine these requests into an ordered set of transactions, call the ZK prover to generate the proofs and the ZK rollup to roll them up, and post the resulting proof to the consensus layer and the set of transactions to the data availability layer. The Casper Association is considering building an API Gateway in front of the L2 nodes in order to assure good functioning of the second layer and a reasonable distribution of work.

8 MVP

Now that we discussed the general structure of any L2 ZKR, we can dig into this project's MVP form. The goal of the MVP is to launch a full ACTUS-based L2 ZKR on top of Casper while promoting privacy. We want to allow financial institutions to log their financial contracts on Casper right from the MVP's launch. Within these constraints, we want to centralize the service as much as needed in order to keep the project's scope feasible, taking into account the Casper Association's resources. The proposed solution to such optimization problem is as follows:

- Contracts: ACTUS implementation in Rust
- ZK prover: Turn ACTUS contract initialization and state updates into ZKPs using one ZK prover system
- Consensus layer: Intergate with Casper's L1
- Data availability layer: Centralized append-only dB, read-only acces-

sible through a RESTful API

- Rollup: Compress ZKPs
- L2 nodes: The MVP will consist of a single node, both built, deployed and maintained by Casper

The ZK prover and ACTUS code will be available both to the single L2 node and any clients, so they can choose to generate the ZKPs themselves, thereby preserving their privacy in respect to Casper itself.

@Mark Is this a reasonable representation of the MVP?

9 Long-term vision

In the long-term, ACTUS on Casper should be sufficiently easy to use and sufficiently decentralized that financial institutions can replace their core software with it (e.g. mainframes), and that any risk analyst or regulator can run analyses and simulations of their choice quickly.

This vision requires two main improvements: The UX for customers and analysts, and increase decentralization.

- Build an SDK to generate ZKPs for your ACTUS contracts and submit both proven and raw contracts to the L2.
- Build a tool to collect all ACTUS contracts and their current state from a given financial institution.
- Decentralize the data availability layer, e.g. by working with IPFS.
- Decentralize the L2. How does consensus work here?

@Mark Is this a reasonable representation of the long-term vision? Should something be mentioned about the API gateway?

@Mark Should we mention something about “if you don’t include something new and interesting in the Casper blockchain, there’s no reason for anyone to adopt Casper”?

10 Conclusions

In this essay, we discussed implementing ACTUS on the Casper blockchain. In the process, we dug through two major new innovations: The ACTUS standard on financial contracts and zero knowledge proofs. We linked everything together into an overview of how to integrate ACTUS with the Casper chain, and explored why one would be interested in doing so. As it turns out, Casper is in need of a big innovation to create an explicit selling point for a given audience, growing its userbase.

@Mark: Do you want to add a last paragraph here, with some call to action or such?