Decentralized Finance: On Blockchain- and Smart Contract-based Financial Markets

Fabian Schär

The term decentralized finance (DeFi) refers to an alternative financial infrastructure built on top of the Ethereum blockchain. DeFi uses smart contracts to create protocols that replicate existing financial services in a more open, interoperable, and transparent way. This paper highlights opportunities and potential risks of the DeFi ecosystem. I propose a multi-layered framework to analyze the implicit architecture and the various DeFi building blocks, including token standards, decentralized exchanges, decentralized debt markets, blockchain derivatives, and on-chain asset management protocols. I conclude that DeFi still is a niche market with certain risks but that it also has interesting properties in terms of efficiency, transparency, accessibility, and composability. As such, DeFi may potentially contribute to a more robust and transparent financial infrastructure. (JEL G15, G23, E59)

First Version: 4. May 2020
This Version: 4. February 2021

Citation info:
This is the working paper. Please cite the journal version – available here.

1 INTRODUCTION

Decentralized finance (DeFi) is a blockchain-based financial infrastructure that has recently gained a lot of traction. The term generally refers to an open, permissionless, and highly interoperable protocol stack built on public smart contract platforms, such as the Ethereum blockchain (see Buterin, 2013). It replicates existing financial services in a more open and transparent way. In particular, DeFi does not rely on intermediaries and centralized institutions. Instead, it is based on open protocols and decentralized applications (DApps). Agreements are enforced by code, transactions are executed in a secure and verifiable way, and legitimate state changes persist on a public blockchain. Thus, this architecture can create an immutable and highly interoperable financial system with unprecedented transparency, equal access rights, and little need for custodians, central clearing houses, or escrow services, as most of these roles can be assumed by "smart contracts."

DeFi already offers a wide variety of applications. For example, one can buy U.S. dollar (USD)-pegged assets (so-called stablecoins) on decentralized exchanges, move these assets to an equally decentralized lending platform to earn interest, and subsequently add the interest-bearing instruments to a decentralized liquidity pool or an on-chain investment fund.

The backbone of all DeFi protocols and applications is smart contracts. Smart contracts generally refer to small applications stored on a blockchain and executed in parallel by a large set of validators. In the context of public blockchains, the network is designed so that each participant can be involved in and verify the correct execution of any operation. As a result, smart contracts are somewhat inefficient compared with traditional centralized computing. However, their advantage is a high level of security: Smart contracts will always be executed as specified and allow anyone to verify the resulting state changes independently. When implemented securely, smart contracts are highly transparent and minimize the risk of manipulation and arbitrary intervention.

To understand the novelty of smart contracts, we first must look at regular server-based web applications. When a user interacts with such an application, they cannot observe the application's internal logic. Moreover, the user is not in control of the execution environment. Either one (or both) could be manipulated. As a result, the user has to trust the application service provider. Smart contracts mitigate both problems and ensure that an application runs as expected. The contract code is stored on the underlying blockchain and can therefore be publicly scrutinized. The contract's behavior is deterministic, and function calls (in the form of transactions) are processed by thousands of network participants in parallel, ensuring the execution's legitimacy. When the execution leads to state changes, for example, the change of account balances, these changes are subject to the blockchain network's consensus rules and will be reflected in and protected by the blockchain's state tree.

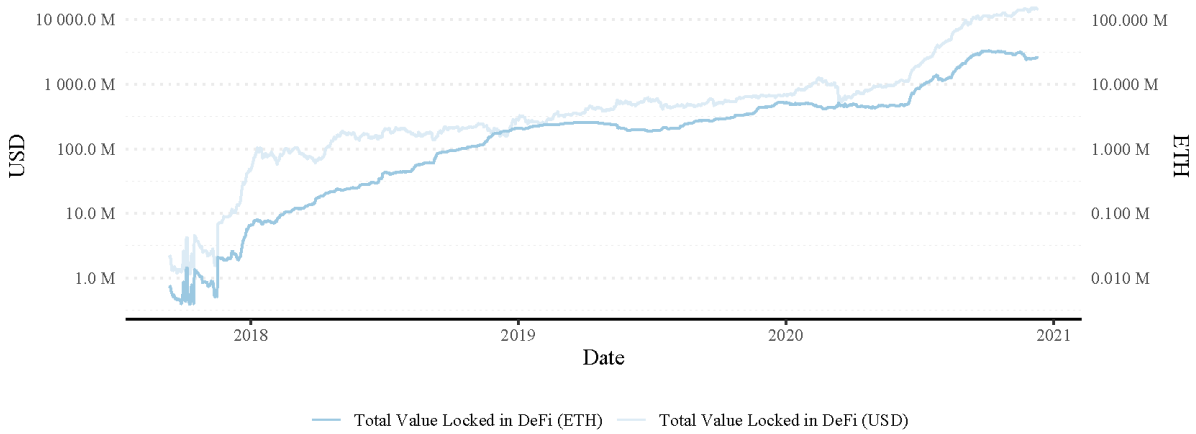
Smart contracts have access to a rich instruction set and are therefore quite flexible. Additionally, they can store cryptoassets and thereby assume the role of a custodian, with entirely customizable criteria for how, when, and to whom these assets can be released. This allows for a large variety of novel applications and flourishing ecosystems.

The original concept of a smart contract was coined by Szabo (1994). Szabo (1997) used the example of a vending machine to describe the idea further and argued that many agreements could be "embedded in the hardware and software we deal with, in such a way as to make a breach of contract expensive…for the breacher." Buterin (2013) proposed a decentralized blockchain-based smart contract platform to solve any trust issues regarding the execution environment and to enable secure global states. Additionally, this platform allows the contracts to interact with and build on top of each other (composability). The concept was further formalized by Wood (2015) and implemented under the name Ethereum. Although there are many alternatives, Ethereum is the largest smart contract platform in terms of market cap, available applications, and development activity.

DeFi still is a niche market with relatively low volumes—however, these numbers are growing rapidly. The value of funds that are locked in DeFi-related smart contracts recently crossed 10 billion USD. It is essential to understand that these are not transaction volume or market cap numbers; the value refers to reserves locked in smart contracts for use in various ways that will be explained in the course of this paper. Figure 1 shows the Ether (ETH, the native cryptoasset of Ethereum) and USD values of the assets locked in DeFi applications.

Figure 1

Total Value Locked in DeFi Contracts (USD and ETH)



Data Source: DeFi Pulse

The spectacular growth of these assets alongside some truly innovative protocols suggests that DeFi may become relevant in a much broader context and has sparked interest among policymakers, researchers, and financial institutions. This article is targeted at individuals from these organizations with an economics or legal background and serves as a survey and an introduction to the topic. In particular, it identifies opportunities and risks and should be seen as a foundation for further research.

2 DEFI BUILDING BLOCKS

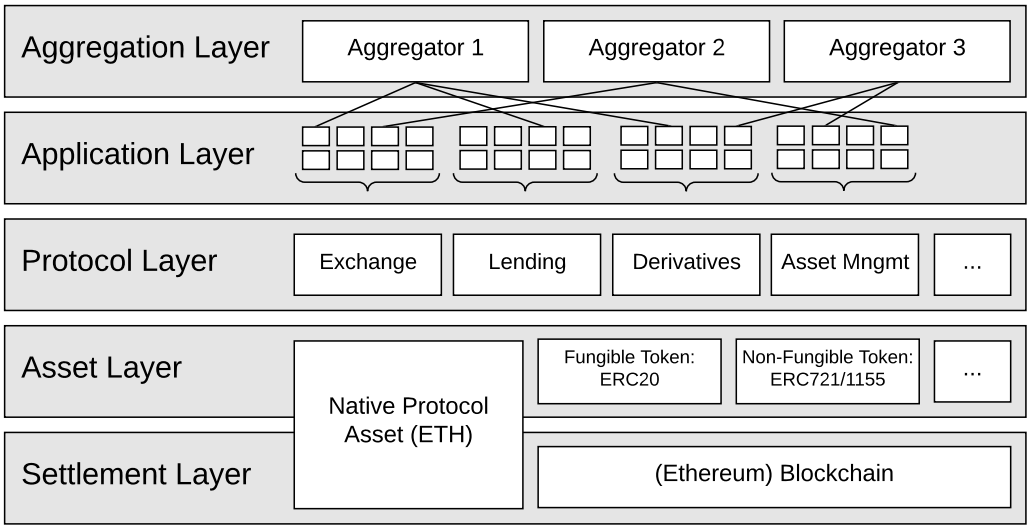
DeFi uses a multi-layered architecture. Every layer has a distinct purpose. The layers build on each other and create an open and highly composable infrastructure that allows everyone to build on, rehash, or use other parts of the stack. It is also crucial to understand that these layers are hierarchical: They are only as secure as the layers below. If, for example, the blockchain in the settlement layer is compromised, all subsequent layers would not be secure. Similarly, if we were to use a permissioned ledger as the foundation, any decentralization efforts on subsequent layers would be ineffective.

This section proposes a conceptual framework for analyzing these layers and studying the token and the protocol layers in greater detail.[1] It differentiates between five layers, as shown in Figure 2: the *settlement*, *asset*, *protocol*, *application,* and *aggregation* layers.

1. The *settlement layer* (Layer 1) consists of the blockchain and its native protocol asset (e.g., Bitcoin [BTC] on the Bitcoin blockchain and ETH on the Ethereum blockchain). It allows the network to store ownership information securely and ensures that any state changes adhere to its ruleset. The blockchain can be seen as the foundation for trustless execution and serves as a settlement and dispute resolution layer.

2. The *asset layer* (Layer 2) consists of all assets that are issued on top of the settlement layer. This includes the native protocol asset as well as any additional assets that are issued on this blockchain (usually referred to as tokens).
3. The *protocol layer* (Layer 3) provides standards for specific use cases such as decentralized exchanges, debt markets, derivatives, and on-chain asset management. These standards are usually implemented as a set of smart contracts and can be accessed by any user (or DeFi application). As such, these protocols are highly interoperable.
4. The *application layer* (Layer 4) creates user-oriented applications that connect to individual protocols. The smart contract interaction is usually abstracted by a web browser-based front end, making the protocols easier to use.
5. The *aggregation layer* (Layer 5) is an extension of the application layer. Aggregators create user-centric platforms that connect to several applications and protocols. They usually provide tools to compare and rate services, allow users to perform otherwise complex tasks by connecting to several protocols simultaneously, and combine relevant information in a clear and concise manner.

Figure 2
The DeFi Stack



Now that we understand the conceptual model, let us take a closer look at tokenization and the protocol layer. After a short introduction to asset tokenization, we will investigate decentralized exchange protocols, decentralized lending platforms, decentralized derivatives, and on-chain asset management. This allows us to establish the foundation needed for our analysis of the potential and risks of DeFi.[2]

2.1 Asset Tokenization

Public blockchains are databases that allow participants to establish a shared and immutable record of ownership—a ledger. Usually, a ledger is used to track the native protocol asset of the respective blockchain. However, when public blockchain technology

became more popular, so did the idea of making additional assets available on these ledgers. The process of adding new assets to a blockchain is called tokenization, and the blockchain representation of the asset is referred to as a token.

The general idea of tokenization is to make assets more accessible and transactions more efficient. In particular, tokenized assets can be transferred easily and within seconds from and to anyone in the world. They can be used in many decentralized applications and stored within smart contracts. As such, these tokens are an essential part of the DeFi ecosystem.

From a technological perspective, there are various ways in which public blockchain tokens can be created (see Roth, Schär, and Schöpfer, 2019). However, most of these options can be ignored, as the vast majority of tokens are issued on the Ethereum blockchain through a smart contract template referred to as the ERC-20 token standard (Vogelsteller and Buterin, 2015). These tokens are interoperable and can be used in almost all DeFi applications. As of January 2021, there are over 350,000 ERC-20 token contracts deployed on Ethereum. [3] Table 1 shows the number of tokens listed on exchanges and the aggregated token market cap in USD per blockchain. Almost 90 percent of all listed tokens are issued on the Ethereum blockchain. The slight deviation in terms of market cap originates from the fact that a relatively large portion of the USDT stablecoin has been issued on Omni.

Table 1

Listed Tokens and Total Token Market Cap by Blockchain Platform

	Number		Market Capitalization (USD)	
Platform	**Absolute**	**Relative**	**Absolute**	**Relative**
Ethereum	1,793	86.74%	55,071,650,000	85.55%
TRON	26	1.26%	4,639,184,120	7.21%
Binance Chain	83	4.02%	2,297,032,000	3.57%
Omni	3	0.15%	1,407,629,950	2.19%
Neo	25	1.21%	160,789,200	0.25%
XRP	1	0.05%	156,223,800	0.24%
Stellar	21	1.02%	155,640,200	0.24%
EOS	31	1.50%	117,560,200	0.18%
Qtum	8	0.39%	71,898,580	0.11%
RSK Smart Bitcoin	1	0.05%	70,715,650	0.11%
Others	75	3.63%	227,652,769	0.35%

Data sources: coinmarketcap.com and tether.to per September 3rd, 2020. Data preparation in the style of Roth et al. (2019).

From an economic perspective, I am more interested in the asset's nature than in the underlying technical standard used to implement the asset's digital representation. The main motivation for adding additional assets on-chain is the addition of a stablecoin. While it would be possible to use the aforementioned protocol assets (BTC or ETH), many

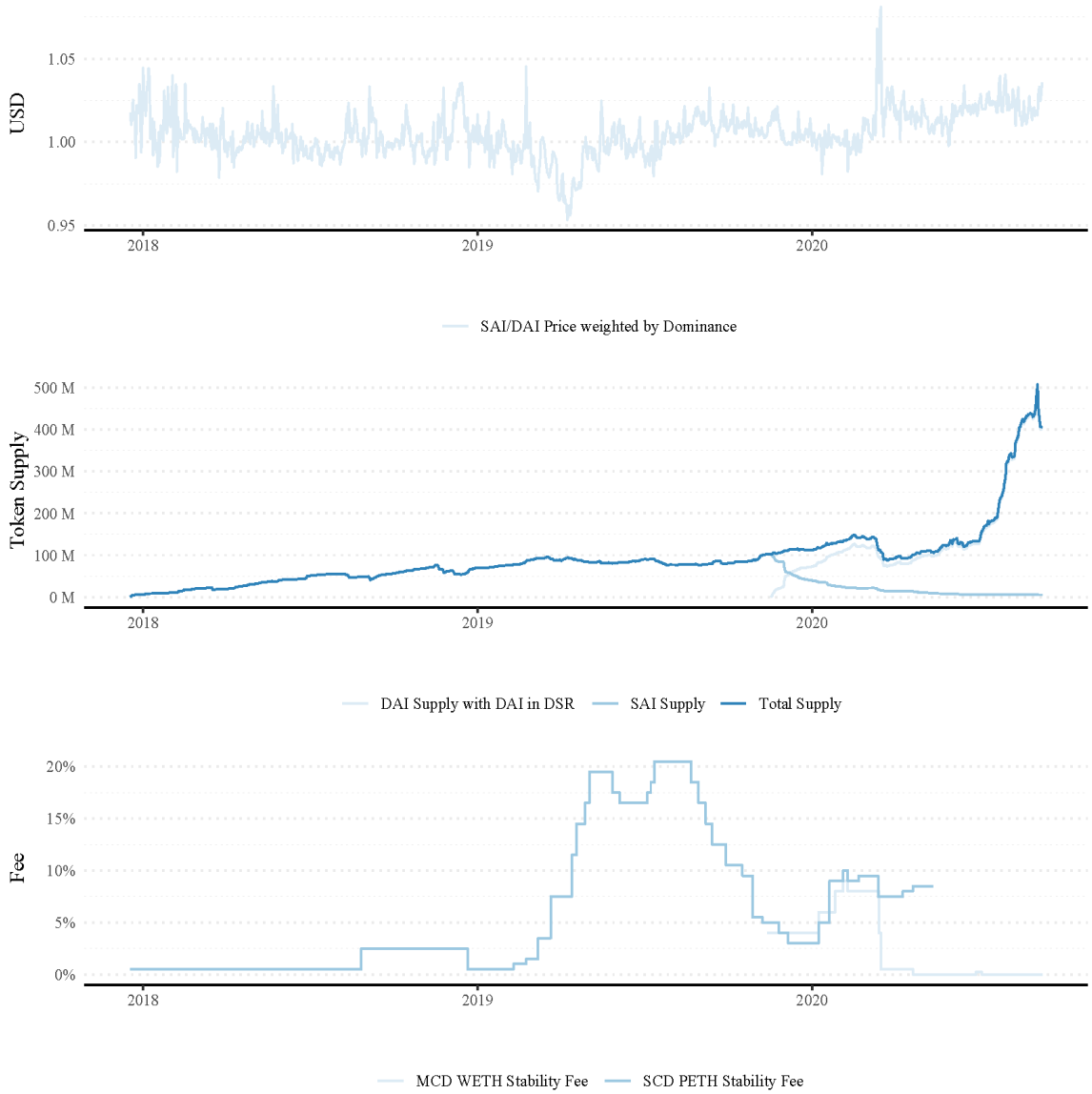
financial contracts require a low-volatility asset. Tokenization enables the creation of these assets.

However, one of the main concerns with tokenized assets is issuer risk. Native digital tokens, such as BTC and ETH, are unproblematic in this regard. In contrast, when someone introduces tokens with a promise, for example, interest payments, dividends, or the delivery of a good or service, the corresponding token's value will depend on this claim's credibility. If an issuer is unwilling or unable to deliver, the token may become worthless or trade at a significant discount. This logic also applies to stablecoins.

Generally speaking, there are three backing models for promise-based tokens: *off-chain collateral*, *on-chain collateral*, and *no collateral*. Off-chain collateral means that the underlying assets are stored with an escrow service, for example, a commercial bank. On-chain collateral means that the assets are locked on the blockchain, usually within a smart contract.[4] When there is no collateral, counterparty risk is at its highest. In this case, the promise is entirely trust-based. Berentsen and Schär (2019) have analyzed the three categories in the context of stablecoins.

On-chain collateral has several advantages. It is highly transparent, and claims can be secured by smart contracts, allowing processes to be executed in a semi-automatic way. A disadvantage of on-chain collateral is that this collateral is usually held in a native protocol asset (or a derivative thereof) and, therefore, will experience price fluctuations. Take the example of the Dai stablecoin, which mainly uses ETH as its on-chain collateral to create a decentralized and trustless Dai token pegged to the value of 1 USD. Since there is no native USD-pegged token on Ethereum, Dai tokens must be backed by another asset. Whenever anyone wants to issue new Dai tokens, they first need to lock enough ETH as underlying collateral in a smart contract provided by the Maker Protocol. Since the USD/ETH exchange rate is not fixed, there is a need for over-collateralization. If the value of the underlying ETH collateral at any point falls below the minimum threshold of 150 percent of the outstanding Dai value, the smart contract will auction off the collateral to cancel the debt in Dai.

Figure 3
Dai Stablecoin Key Metrics



Data Sources: DeFi Pulse, Coinmarketcap

Figure 3 shows some key metrics of the Dai stablecoin, including price, total Dai in circulation, and the stability fee, that is, the interest rate that has to be paid by anyone who is creating new Dai (see Section 2.3).

There are also several examples of off-chain collateralized stablecoins. The most popular ones are USDT and USDC, both USD-backed stablecoins. They are both available as ERC-20 tokens on the Ethereum blockchain. DGX is an ERC-20 based stablecoin backed by gold, and WBTC is a tokenized version of Bitcoin, making Bitcoin available on the Ethereum blockchain. Off-chain collateralized tokens can mitigate exchange rate risk, as the collateral may be equivalent to the tokenized claim (e.g., USD claim, backed by real USD). However, off-chain collateralized tokens introduce counterparty risk and external dependencies. Tokens that use off-chain collateral require regular audits and precautionary

measures to ensure that the underlying collateral is available at all times. This process is costly and, in many cases, not entirely transparent for the token holders.

While I am unaware of any functional designs for unbacked stablecoins, that is, stablecoins that do not use any form of collateral to maintain the peg, several organizations are working on that idea. Note that rebase tokens such as Ampleforth or YAM do not qualify as stablecoins. They only provide a stable unit of account but still expose the holder to volatility in the form of a dynamic token quantity.

Although stablecoins serve a vital role in the DeFi ecosystem, it would not do justice to the subject of tokenization to limit the discussion to these assets. There are all kinds of tokens that serve a variety of purposes, including governance tokens for decentralized autonomous organizations (DAO), tokens that allow the holder to perform specific actions in a smart contract, tokens that resemble shares or bonds, and even synthetic tokens that can track the price of any real-world asset.

Another distinct category are so-called non-fungible tokens (NFTs). NFTs are tokens that represent unique assets, that is, collectibles. They can either be the digital representation of a physical object such as a piece of art, making them subject to the usual counterparty risk, or a digitally native unit of value with unique characteristics. In any case, the token's non-fungibility attributes ensure that the ownership of each asset can be individually tracked and the asset precisely identified. NFTs usually are built on the ERC-721 token standard (Entriken et al., 2018).

The following sections discuss the protocol layer and examine how tokens can be traded using decentralized exchanges (Section 2.2), how they can be used as collateral for loans (Section 2.3) and to create decentralized derivatives (Section 2.4), and how they can be included in on-chain investment funds (Section 2.5).

2.2 Decentralized Exchange Protocols

As of September 2020, there are over 7,092 cryptoassets[5] listed on exchanges. While most of them are economically irrelevant and have a negligible market cap and trading volume, there is a need for marketplaces where people can trade the more popular ones. This would allow owners of such assets to rebalance their exposure according to their preferences and risk profiles and adjust portfolio allocations.

In most cases, cryptoasset trades are conducted through centralized exchanges. Centralized exchanges are relatively efficient, but they have one severe problem. To be able to trade on a centralized exchange, traders must first deposit assets with the exchange. They thereby forfeit direct access to their assets and have to trust the exchange operator. Dishonest or unprofessional exchange operators may confiscate or lose assets. Moreover, centralized exchanges create a single point of attack and face the constant threat of becoming the target of malicious third parties. The relatively low regulatory scrutiny intensifies both problems and the immense scaling efforts many of these exchanges had to go through within a short time. Accordingly, it is no surprise that some centralized cryptoasset exchanges have lost customer funds.

Decentralized exchange protocols try to mitigate these issues by removing the trust requirement. Users no longer must deposit their funds with a centralized exchange.

Instead, they remain in exclusive control of their assets until the trade is executed. Trade execution happens atomically through a smart contract, meaning that both sides of the trade are performed in one indivisible transaction, mitigating the counterparty credit risk. Depending on the exact implementation, the smart contract may assume additional roles, effectively making many intermediaries such as escrow services and central counterparty clearing houses (CCPs) obsolete.

Early decentralized exchanges such as EtherDelta have been set up as walled gardens with no interaction between the various implementations. The exchanges had no shared liquidity, leading to relatively low transaction volumes and large bid/ask spreads. High network fees, as well as cumbersome and slow processes to move funds between these decentralized exchanges, have rendered supposed arbitrage opportunities useless.

More recently, there has been a move toward open exchange protocols. These projects try to streamline the architecture of decentralized exchanges by providing standards on how asset exchange can be conducted and allowing any exchange built on top of the protocol to use shared liquidity pools and other protocol features. However, most importantly, other DeFi protocols can use these marketplaces and exchange or liquidate tokens when needed.

In the following subsections, I compare various types of decentralized exchange protocols, some of which are not exchanges in the narrow sense but have been included in the analysis, as they serve the same purpose. The results are summarized in Table 2.

Table 2
Most Popular Decentralized Exchange Protocols

Protocol Name	Protocol Type	Price Discovery
0x	Exchange	Off-Chain Order Books
(Air)Swap	P2P / OTC	P2P Negotiation
Bancor	Constant Function Market Maker	Smart Contract
Balancer	Constant Function Market Maker	Smart Contract
Curve	Constant Function Market Maker	Smart Contract
Kyber Network	Reserve Aggregator	Proposal by Maker
UniSwap	Constant Function Market Maker	Smart Contract

Decentralized Order Book Exchanges

Decentralized order book exchanges can be implemented in a variety of ways. They all use smart contracts for transaction settlement, but they differ significantly in how the order books are hosted. One has to distinguish between on-chain and off-chain order books.

On-chain order books have the advantage of being entirely decentralized. Every order is stored within the smart contract. As such, there is no need for additional infrastructure or third-party hosts. The disadvantage of this approach is that every action requires a blockchain transaction. Therefore, it is a costly and slow process for which even the

declaration of the intent to trade results in network fees. Considering that volatile markets will require frequent order cancellations, this disadvantage becomes even more costly.

For this reason, many decentralized exchange protocols rely on off-chain order books and only use the blockchain as a settlement layer. Off-chain order books are hosted and updated by centralized third parties, usually referred to as relayers. They provide takers with the information they need to select an order they would like to match. While this approach indeed introduces some centralized components and dependencies to the system, the relayers' role is limited. Relayers are never in control of the funds and neither match nor execute the orders. They simply provide ordered lists with quotes and may charge a fee for that service. The openness of the protocol ensures that there is competition among the relayers and mitigates potential dependencies.

The dominant protocol that uses this approach is called 0x (Warren and Bandeali, 2017). The protocol uses a three-step process for trades. First, the maker sends a pre-signed order to the relayer for inclusion in the order book. Second, a potential taker queries the relayer and selects one of the orders. Third, the taker signs and submits the order to the smart contract, triggering the atomic exchange of the cryptoassets.

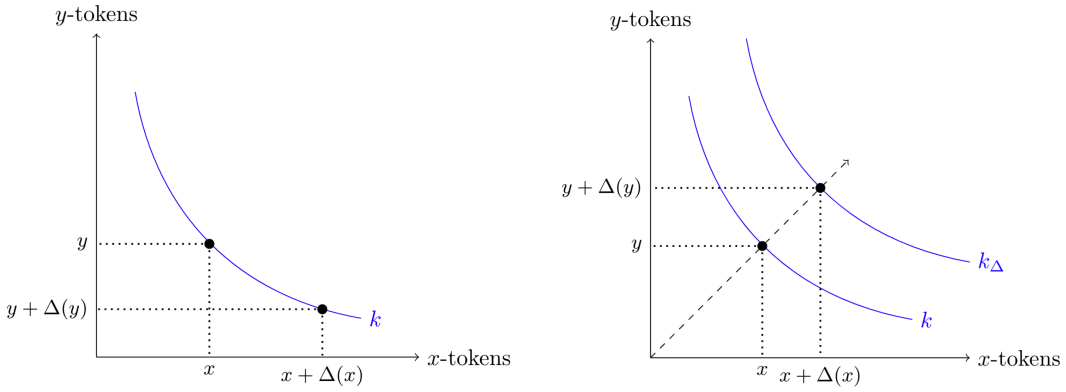
Constant Function Market Maker

A constant function market maker (CFMM) is a smart contract-liquidity pool that holds (at least) two cryptoassets in reserve and allows anyone to deposit tokens of one type and thereby to withdraw tokens of the other type. To determine the exchange rate, smart contract-based liquidity pools use variations of the constant product model, where the relative price is a function of the smart contract's token reserve ratio. The earliest implementation I am aware of was proposed by Hertzog, Benartzi, and Benartzi (2017). Adams (2018) has simplified the model, and Zhang, Chen, and Park (2018) provide a formal proof of the concept. Martinelli and Mushegian (2019) generalized the concept for cases with more than two tokens and dynamic token weights. Egorov (2019) optimized the idea for stablecoin swaps.

In its simplest form, the constant product model can be expressed as $xy = k$, where x and y correspond to the smart contract's token reserves and k is a constant. Considering that this equation must hold, when someone executes a trade, we get $(x + \Delta x) \cdot (y + \Delta y) = k$. It can then be easily shown that $\Delta y = \frac{k}{x+\Delta x} - y$. Consequently, Δy will assume negative values for any $\Delta x > 0$. In fact, any exchange corresponds to a move on a convex token reserve curve, which is shown in Figure 4A. A liquidity pool using this model cannot be depleted, as tokens will get more expensive with lower reserves. When the token supply of either one of the two tokens approaches zero, its relative price rises infinitely as a result.

Figure 4

Visualization of Liquidity Pool Token Reserves in a Constant Product Model



It is important to point out that smart contract-based liquidity pools are not reliant on external price feeds (so-called oracles). Whenever the market price of an asset shifts, anyone can use the arbitrage opportunity and trade tokens with the smart contract until the liquidity pool price converges to the current market price. The implicit bid/ask spread of the constant product model (plus a small trading fee) may lead to the accumulation of additional funds. Anyone who provides liquidity to the pool receives pool share tokens that allow them to participate in this accumulation and to redeem these tokens for their share of a potentially growing liquidity pool. Liquidity provision results in a growing k and is visualized in Figure 4B.

Prominent examples of smart contract-based liquidity pool protocols are UniSwap, Balancer, Curve, and Bancor.

Smart Contract-Based Reserve Aggregation

Another approach is to consolidate liquidity reserves through a smart contract that allows large liquidity providers to connect and advertise prices for specific trade pairs. A user who wants to exchange token x for token y may send a trade request to the smart contract. The smart contract will compare prices from all liquidity providers, accept the best offer on behalf of the user, and execute the trade. It acts as a gateway between users and liquidity providers, ensuring best execution and atomic settlement.

In contrast to smart contract-based liquidity pools, with smart contract-based reserve aggregation, prices are not determined within the smart contract. Instead, prices are set by the liquidity providers. This approach works fine if there is a relatively broad base of liquidity providers. However, if there is limited or no competition for a given trade pair, the approach may result in collusion risks or even monopolistic price setting. As a countermeasure, reserve aggregation protocols usually have some (centralized) control mechanisms, such as maximum prices or a minimum number of liquidity providers. In some cases, liquidity providers may only participate after a background check, including KYC (know your customer) verification.

The best-known implementation of this concept is the Kyber Network (Luu and Velner, 2017), which serves as a backbone protocol for a large variety of DeFi applications.

Peer-to-Peer Protocols

An alternative to classic exchange or liquidity pool models are peer-to-peer (P2P) protocols, also called over-the-counter (OTC) protocols. They mostly rely on a two-step approach, where participants can query the network for counterparties who would like to trade a given pair of cryptoassets and then negotiate the exchange rate bilaterally. Once the two parties agree on a price, the trade is executed on-chain via a smart contract. In contrast to other protocols, offers can be accepted exclusively by the parties who have been involved in the negotiation. In particular, it is not possible for a third party to front-run someone accepting an offer by observing the pool of unconfirmed transactions (mempool).

To make things more efficient, the process is usually automated. Additionally, one can use off-chain indexers for peer discovery. These indexers assume the role of a directory in which people can advertise their intent to make a specific trade. Note that these indexers only serve to establish a connection. Prices are still negotiated P2P.

AirSwap is the most popular implementation of a decentralized P2P protocol. It was proposed by Oved and Mosites (2017).

2.3 Decentralized Lending Platforms

Loans are an essential part of the DeFi ecosystem. There are a large variety of protocols that allow people to lend and borrow cryptoassets. Decentralized loan platforms are unique in the sense that they require neither the borrower nor the lender to identify themselves. Everyone has access to the platform and can potentially borrow money or provide liquidity to earn interest. As such, DeFi loans are completely permissionless and not reliant on trusted relationships.

To protect the lender and stop the borrower from running away with the funds, there are two distinct approaches:

First, credit can be provided under the condition that the loan must be repaid atomically, meaning that the borrower receives the funds, uses, and repays them—all within the same blockchain transaction. Suppose the borrower has not returned the funds (plus interest) at the end of the transaction's execution cycle. In this case, the transaction will be invalid and any of its results (including the loan itself) reverted. These so-called flash loans (Wolff, 2018; Boado, 2020) are an exciting but still highly experimental application. While flash loans can only be employed in applications that are settled atomically and entirely on-chain, they are an efficient new instrument for arbitrage and portfolio restructuring. As such, they are on track to become an essential part of DeFi lending.

Second, loans can be fully secured with collateral. The collateral is locked in a smart contract and only released once the debt is repaid. Collateralized loan platforms exist in three variations: *Collateralized debt positions*, *pooled collateralized debt markets,* and *P2P collateralized debt markets*. Collateralized debt positions are loans that use newly created tokens, while debt markets use existing tokens and require a match between a borrowing and a lending party. The three variations are discussed below.

Collateralized Debt Positions

Some DeFi applications allow users to create collateralized debt positions and thereby issue new tokens that are backed by the collateral. To be able to create these tokens, the person must lock cryptoassets in a smart contract. The number of tokens that can be created depends on the target price of the tokens generated, the value of the cryptoassets that are being used as collateral, and the target collateralization ratio. The newly created tokens are essentially fully collateralized loans that do not require a counterparty and allow the user to get a liquid asset while maintaining market exposure through the collateral. The loan can be used for consumption, allowing the person to overcome a temporary liquidity squeeze or to acquire additional cryptoassets for leveraged exposure.

To illustrate the concept, let us use the example of MakerDAO, a decentralized protocol that is used to issue the USD-pegged Dai stablecoin. First, the user deposits ETH in a smart contract classified as a collateralized debt position (CDP) (or vault). Subsequently, they call a contract function to create and withdraw a certain number of Dai and thereby lock the collateral. This process currently requires a minimum collateralization ratio of 150 percent, meaning that for any 100 USD of ETH locked up in the contract, the user can create at most 66.66 Dai.[6]

Any outstanding Dai is subject to a stability fee, which in theory should correspond to the Dai debt market's maximum interest rate. This rate is set by the community, namely the MKR token holders. MKR is the governance token for the MakerDAO project. As shown in Figure 3, the stability fee has been fluctuating wildly between 0 and 20 percent.

To close a CDP, the owner must send the outstanding Dai plus the accumulated interest to the contract. The smart contract will allow the owner to withdraw their collateral once the debt is repaid. If the borrower fails to repay the debt, or if the collateral's value falls below the 150 percent threshold, where the full collateralization of the loan is at risk, the smart contract will start to liquidate the collateral at a potentially discounted rate.

Interest payments and liquidation fees are partially used to "burn" MKR, thereby decreasing the total MKR supply. In exchange, MKR holders assume the residual risk of extreme negative ETH price shocks, which may lead to a situation in which the collateral is insufficient to maintain the USD peg. In this case, new MKR will be created and sold at a discounted rate. As such, MKR holders have skin in the game, and it should be in their best interest to maintain a healthy system.

It is important to mention that the MakerDAO system is much more complicated than what is described here. Although the system is mostly decentralized, it is reliant on price oracles, which introduce some dependencies, as discussed in Section 3.2.

MakerDAO has recently switched to a multi-collateral system, with the goal to make the protocol more scalable by allowing a variety of cryptoassets to be used as collateral.

Collateralized Debt Markets

Instead of creating new tokens, it is also possible to borrow existing cryptoassets from someone else. For obvious reasons, this approach requires a counterparty with opposing preferences. In other words: For someone to be able to borrow ETH, there must be another person willing to lend ETH. To mitigate counterparty risk and protect the lender, loans must be fully collateralized, and the collateral is locked in a smart contract—just as in our previous example.

Matching lenders with borrowers can be done in a variety of ways. The broad categories are P2P and pooled matching. P2P matching means that the person who is providing the liquidity lends the cryptoassets to specific borrowers. Consequently, the lender will only start to earn interest once there is a match. The advantage of this approach is that the parties agree on a time period and operate with fixed interest rates.

Pooled loans use variable interest rates that are subject to supply and demand. The funds of all borrowers are aggregated in a single, smart contract-based lending pool, and lenders start to earn interest right when they deposit their funds in the pool. However, the interest rates are a function of the pool's utilization rate. When liquidity is readily available, loans will be cheap. When it is in great demand, loans will become more expensive. Lending pools have the additional advantage that they can perform maturity and size transformation while maintaining relatively high liquidity for the individual lender.

There is a large variety of lending protocols. Some of the most popular ones are Aave (Boado, 2020), Compound (Leshner and Hayes, 2019), and dYdX (Juliano, 2017). Figure 5 shows the asset-weighted borrowing and lending rates for Dai and ETH. For Dai, the figure also includes the MakerDAO stability fee, which should always be the highest rate in the system. Surprisingly, this is not always the case, meaning that some people have paid a price premium in the secondary market. As of September 2020, Dai accounts for almost 75 percent of all loans in the DeFi ecosystem.

Figure 5

Weighted Dai Collateralized Debt Market Rates and MakerDAO Stability Fee

Data Source: DeFi Pulse

2.4 Decentralized Derivatives

Decentralized derivatives are tokens that derive their value from an underlying asset's performance, the outcome of an event, or the development of any other observable variable. They usually require an oracle to track these variables and therefore introduce some dependencies and centralized components. The dependencies can be reduced when the derivative contract uses multiple independent data sources.

We differentiate between asset-based and event-based derivative tokens. We call a derivative token asset-based when its price is a function of an underlying asset's performance. We call a derivative event-based when its price is a function of any observable variable that is not the performance of an asset. Both categories will be discussed in the following sections.

Asset-Based Derivative Tokens

Asset-based derivative tokens are an extension of the CDP model described in Section 2.3. Instead of limiting the issuance to USD-pegged stablecoins, the locked collateral can be used to issue synthetic tokens that follow the price movements of a variety of assets. Examples include tokenized versions of stocks, precious metals, and alternative cryptoassets. The higher the underlying volatility, the larger the risk of falling below a given collateralization ratio.

A popular derivative token platform is called Synthetix (Brooks et al., 2018). It is implemented so that the total debt pool of all participants increases or decreases depending on the aggregate price of all outstanding synthetic assets. This ensures that tokens with the same underlying assets remain fungible; that is, redemption does not depend on the issuer. The flip side of this design is that users assume additional risk when they mint assets, as their debt position will also be affected by everyone else's asset allocation.

A particular case of asset-based derivative tokens are inverse tokens. Here, the price is determined by an inverse function of the underlying assets' performance within a given price range. These inverse tokens allow users to get short exposure to cryptoassets.

Event-Based Derivative Tokens

Event-based derivative tokens can be based on any objectively observable variable with a known set of potential outcomes, a specified observation time, and a resolution source.[7] Anyone can buy a full set of sub-tokens for a given event by locking 1 ETH in a smart contract. A complete set of sub-tokens consists of 1 sub-token for each potential outcome. These sub-tokens can be traded individually. When the market resolves, the smart contract's cryptoassets will be split among the sub-token owners of the winning outcome. In the absence of market distortions, each sub-token's ETH price should, therefore, correspond to the probability of the underlying outcome.

Under certain circumstances, these prediction markets may serve as decentralized oracles for the likelihood of a future outcome. However, market resolution (and therefore the price) greatly depends on the trustworthiness of the resolution source. As such, event-based derivative tokens introduce external dependencies and may be unilaterally influenced by a malicious reporter. Potential attack vectors include flawed or misleading

question specifications, incomplete outcome sets that may render the event unresolvable, and the choice of unreliable or fraudulent resolution sources.

The most popular implementation is called Augur (Peterson et al., 2019). It uses a multi-stage resolution and disputing process that should minimize the dependency on a single reporting source as much as possible. If the token holders do not agree with the designated reporter, they may start a dispute, which should eventually lead to the correct outcome.

2.5 On-Chain Asset Management

Just like traditional investment funds, on-chain funds are mainly used for portfolio diversification. They allow users to invest in a basket of cryptoassets and employ a variety of strategies without having to handle the tokens individually. In contrast to traditional funds, the on-chain variant does not require a custodian. Instead, the cryptoassets are locked up in a smart contract. The investors never lose control over their funds, can withdraw or liquidate them, and can observe the smart contracts' token balances at any point in time.

The smart contracts are set up in such a way that they follow a variety of simple strategies, including semi-automatic rebalancing of portfolio weights and trend trading, using moving averages. Alternatively, one or multiple fund managers can be selected to manage the fund actively. In this case, the smart contract ensures that asset managers adhere to the predefined strategy and act in the investors' best interest. In particular, asset managers are limited to actions in accordance with the fund's ruleset and the risk profile stipulated in the smart contract. The smart contract can mitigate many forms of the principal-agent problem and incorporate regulatory requirements by enforcing them on-chain. As a result, on-chain asset management may lead to lower fund setup and auditing costs.

Whenever someone invests in an on-chain fund, the corresponding smart contract issues *fund tokens* and transfers them to the investor's account. These tokens represent partial ownership of the fund and allow token holders to redeem or liquidate their share of the assets. For example, if an investor owns 1 percent of the fund tokens, this person would be entitled to 1 percent of the locked cryptoassets. When the investor decides to close out the investment, the fund tokens get burned, the underlying assets are sold on a decentralized exchange, and the investor is compensated with the ETH-equivalent of their share of the basket.

There are several implementations of on-chain fund protocols, including the Set Protocol (Feng and Weickmann, 2019), Enzyme Finance (formerly Melon) (Trinkler and El Isa, 2017), Yearn Vaults (Cronje, 2020), and Betoken (Liu and Palayer, 2018). All of these implementations are limited to ERC-20 tokens and Ether. Moreover, they heavily depend on price oracles and third-party protocols, mainly for lending, trading, and the inclusion of low-volatility reference assets such as the Dai or USDC stablecoins. Consequently, there are severe dependencies, which will be discussed in Section 3.2.

Both Enzyme Finance and Set Protocol allow anyone to create new investment funds. Enzyme Finance has a focus on building an infrastructure for decentralized funds, using smart contract-based rulesets to ensure that fund managers stick to the funds' strategies. Trading restriction parameters such as maximum concentration, price tolerance, and the maximum number of positions, as well as user and asset whitelists and blacklists, are enforced by these smart contracts. The same is true for the fund's fee schedule. Set Protocol is mainly designed for semi-automated strategies with deterministic portfolio rebalancing triggered by predefined threshold values and timelocks. However, the protocol is also used for active management. Betoken operates as a single fund of funds managed by a community of asset managers through a meritocratic system. The more successful an individual fund manager is, the greater their future influence on allocating the collective resources. UniSwap's liquidity pool (see Section 2.2) also has some characteristics of an on-chain investment fund. The constant product model creates the incentives for a semi-automatic rebalancing of portfolio weights, while the trading fees generate passive income for the investors.

Yearn Vaults are collective investment pools designed to maximize yield for a given asset. Strategies are quite diverse but usually involve several steps and active management. In many cases, these actions would be too expensive (in terms of transaction fees) for smaller amounts. Moreover, they require that the investor is vigilant and well-informed. Yearn Vaults mitigate these issues by employing the knowledge of the masses and using collective action to split network fees proportionally among all participants. However, the deep integration of the protocol also introduces severe dependencies.

3 OPPORTUNITIES & RISKS

In this section, we analyze the opportunities and risks of the DeFi ecosystem. It lays the foundation for the discussion in Section 4.

3.1 Opportunities

DeFi may increase the *efficiency*, *transparency*, and *accessibility* of the financial infrastructure. Moreover, the system's *composability* allows anyone to combine multiple applications and protocols, thereby creating new and exciting services. We discuss these aspects in the following subsections.

Efficiency

While much of the traditional financial system is trust based and dependent on centralized institutions, DeFi replaces some of these trust requirements with smart contracts. The contracts can assume the roles of custodians, escrow agents, and CCPs. For example, if two parties want to exchange digital assets in the form of tokens, there is no need for guarantees from a CCP. Instead, the two transactions can be settled atomically, meaning that either both or neither of the transfers will be executed. This significantly decreases counterparty credit risk and makes financial transactions much more efficient.

Lower trust requirements may come with the additional benefit of reducing regulatory pressure and reducing the need for third-party audits. Similar efficiency gains are possible for almost every area of the financial infrastructure.

Additionally, token transfers are much faster than any of the transfers in the traditional financial system. Transfer speed and transaction throughput can be further increased with Layer 2 solutions, such as sidechains or state- and payment-channel networks.

Transparency

DeFi applications are transparent. All transactions are publicly observable, and the smart contract code can be analyzed on-chain. The observability and deterministic execution allow—at least in theory—an unprecedented level of transparency.

Financial data are publicly available and may potentially be used by researchers and users alike. In the case of a crisis, the availability of historical (and current) data is a vast improvement over traditional financial systems, where much of the information is scattered across a large number of proprietary databases or not available at all. As such, transparency of DeFi applications may allow for the mitigation of undesirable events before they arise and help provide much faster understanding of their origin and potential consequences when they emerge.

Accessibility

By default, DeFi protocols can be used by anyone. As such, DeFi may potentially create a genuinely open and accessible financial system. In particular, the infrastructure requirements are relatively low and the risk of discrimination is almost inexistent due to the lack of identities.

If regulation demands access restrictions, for example, for security tokens, such restrictions can be implemented in the token contracts without compromising the settlement layer's integrity and decentralization properties.

Composability

DeFi protocols are often compared with Lego pieces. The shared settlement layer allows these protocols and applications to interconnect. On-chain fund protocols can make use of decentralized exchange protocols or achieve leveraged positions through lending protocols.

Any two or more pieces can be integrated, forked, or rehashed to create something entirely new. Anything that has been created before can be used by an individual or by other smart contracts. This flexibility allows for an ever-expanding range of possibilities and unprecedented interest in open financial engineering.

3.2 Risks

DeFi also has certain risks, namely, *smart contract execution risk*, *operational security*, and *dependencies* on other protocols and external data. We discuss these aspects in the following subsections.

Smart Contract Execution

While the deterministic and decentralized execution of smart contracts does have its advantages, there is risk that something may go wrong. If there are coding errors, these errors may potentially create vulnerabilities that allow an attacker to drain the smart contract's funds, cause chaos, or render the protocol unusable. Users have to be aware that the protocol is only as secure as the smart contracts underlying it. Unfortunately, the average user will not be able to read the contract code, let alone evaluate its security. While audits, insurance services, and formal verification are partial solutions to this problem, some degree of uncertainty remains.

Similar risks exist in contract execution. Most users do not understand the data payload they are asked to sign as part of transactions and may be misled by a compromised front-end. Unfortunately, there seems to be an inherent trade-off between usability and security. For example, some decentralized blockchain applications will ask for permissions to transfer an infinite number of tokens on behalf of the user—usually to make future transactions more convenient and efficient. Such permission, however, puts the user's funds at risk.

Operational Security

Many DeFi protocols and applications use admin keys. These keys allow a predefined group of individuals (usually the project's core team) to upgrade the contracts and to perform emergency shutdowns. While it is understandable that some projects want to implement these precautionary measures and remain somewhat flexible, the existence of these keys can be a potential problem. If the keyholders do not create or store their keys securely, malicious third parties could get their hands on these keys and compromise the smart contract. Alternatively, the core team members themselves may be malicious or corrupted by significant monetary incentives.

Most projects try to mitigate this risk with multisig and timelocks. Multisig requires *M*-of-*N* keys to execute any of the smart contract's admin functions, and timelocks specify the earliest time at which a transaction can be (successfully) confirmed.

As an alternative, some projects rely on voting schemes, where the respective governance tokens grant their owners the right to vote on the protocol's future. However, in many cases, the majority of governance tokens are held by a small group of people, effectively leading to similar results as with admin keys. Some projects have tried to mitigate this concentration of voting power by rewarding early adopters and users who fulfill specific criteria, which range from simple protocol usage to active participation in the voting process and third-party token staking (yield farming). Nevertheless, even when a launch is perceived as being relatively "fair," the actual distribution often remains highly concentrated.

Governance tokens may lead to undesirable consequences. In fact, a high concentration of power may be even more problematic when these rights are tokenized. In the absence of vesting periods, malicious founders can pull the rug by dumping their entire token holding on a CFMM, causing a massive supply shock and undermining the project's credibility. Moreover, yield farming may lead to *centralization creep* by allowing an already well-established protocol to assume a significant portion of a relatively new

protocol's governance tokens. This may create large meta protocols whose token holders essentially control a considerable portion of the DeFi infrastructure.

Dependencies

As described in Section 3.1, some of the most promising features of the DeFi ecosystem are its openness and composability. These features allow various smart contracts and decentralized blockchain applications to interact with each other and to offer new services based on a combination of existing ones. On the flip side, these interactions introduce severe dependencies. If there is an issue with one smart contract, it may potentially have wide-reaching consequences for multiple applications across the entire DeFi ecosystem. Moreover, problems with the Dai stablecoin or severe ETH price shocks may cause ripple effects throughout the whole DeFi ecosystem.

The problem becomes apparent when illustrated by an example. Let us assume that a person locks ETH as collateral in the MakerDAO contract to issue Dai stablecoins. Let us further assume that the Dai stablecoins are locked in a compound lending smart contract to issue interest-bearing derivative tokens, called cDai. The cDai tokens are subsequently moved to the UniSwap ETH/cDai liquidity pool, along with some ETH, allowing the person to withdraw UNI-cDai tokens representing a share of the liquidity pool. With every additional smart contract, the potential risk of a bug increases. If any of the contracts in the sequence fail, the UNI-cDai tokens could potentially become worthless. These "token on top of a token on top of a token" scenarios, which create wrapper tokens, can entangle projects in such a way that theoretical transparency does not correspond to actual transparency.

External Data

Another point worth mentioning is the fact that many smart contracts are reliant on external data. Whenever a smart contract depends on data that are not natively available on-chain, the data must be provided by external data sources. These so-called oracles introduce dependencies and may, in some cases, lead to heavily centralized contract execution. To mitigate this risk, many projects rely on decentralized oracle networks with a large variety of data provision schemes.

Illicit Activity

A common concern among regulators is that cryptoassets may be used by individuals who want to avoid records and monitoring. While the inherent transparency of DeFi is a deterrent to this use case, the network's pseudonymity may provide some privacy. However, this may not necessarily be a bad thing, and the situation is more complicated than it may seem at first glance. On the one hand, pseudonymity can be abused by actors with dishonest intentions. On the other hand, privacy may be a desirable attribute for some legitimate financial applications. Correspondingly, regulators should act with great care, trying to find reasonable solutions that allow them to step in where necessary without stifling innovation. Moreover, one has to be aware that regulating a decentralized network may not be feasible.

While it is questionable whether regulators can (or should) regulate a decentralized infrastructure, there are two areas that deserve special attention, namely, fiat on- and off-ramps and the decentralization theater.

Fiat on- and off-ramps are the interface to the traditional financial system. Whenever people want to move assets from their bank account to the blockchain-based system or the other way, they have to go through a financial service provider. These financial service providers are regulated and may require background checks on the origin of the funds.

In a similar vein, it is important to differentiate between legitimate decentralized protocols and projects that only claim to be decentralized but are in fact under the exclusive control of an organization or a few individuals. The former may provide exciting new possibilities and remove some dependencies, while the latter may essentially introduce the worst of two worlds, that is, de facto dependencies on a centralized operator with limited supervision. Keeping this in mind, regulators should watch closely and analyze carefully if a given DeFi protocol is indeed decentralized or if the DeFi label is just for show in an attempt to get around regulation.

Scalability

Blockchains face the ultimate trade-off between decentralization, security, and scalability. While the Ethereum blockchain is generally regarded as relatively decentralized and secure, it struggles to keep up with the great demand for block space. Escalating gas prices (transaction fees) and long confirmation times adversely affect the DeFi ecosystem and favor wealthy individuals who can conduct large trades.

Potential solutions to this problem include base-layer sharding, as well as various Layer 2 solutions, such as state channels, ZK (zero knowledge) rollups, and optimistic rollups. However, in many cases, scalability efforts weaken composability and general transaction atomicity—two of DeFi's most prominent features. On the other hand, moving DeFi to a more centralized base layer does not seem to be a reasonable approach either, as it would essentially undermine its main value proposition. Thus, it remains to be seen if a truly decentralized blockchain can keep up with the demand and provide the foundation for an open, transparent, and immutable financial infrastructure.

4 CONCLUSION

DeFi offers exciting opportunities and has the potential to create a truly open, transparent, and immutable financial infrastructure. Because DeFi consists of numerous highly interoperable protocols and applications, every individual can verify all transactions and data is readily available for users and researchers to analyze.

DeFi has unleashed a wave of innovation. On the one hand, developers are using smart contracts and the decentralized settlement layer to create trustless versions of traditional financial instruments. On the other hand, they are creating entirely new financial instruments that could not be realized without the underlying public blockchain. *Atomic*

swaps, *autonomous liquidity pools*, *decentralized stablecoins*, and *flash loans* are just a few of many examples that show the great potential of this ecosystem.

While this technology has great potential, there are certain risks involved. Smart contracts can have security issues that may allow for unintended usage, and scalability issues limit the number of users. Moreover, the term "decentralized" is deceptive in some cases. Many protocols and applications use external data sources and special admin keys to manage the system, conduct smart contract upgrades, or even perform emergency shutdowns. While this does not necessarily constitute a problem, users should be aware that, in many cases, there is much trust involved. However, if these issues can be solved, DeFi may lead to a paradigm shift in the financial industry and potentially contribute toward a more robust, open, and transparent financial infrastructure.

NOTES

[1] An alternative approach can be found here: https://medium.com/pov-crypto/ethereum-the-digital-finance-stack-4ba988c6c14b

[2] For readers who wish to understand the settlement layer better and want to read a general introduction to Blockchain and cryptocurrencies, we recommend Berentsen and Schär (2018).

[3] Source: etherscan.io/tokens, accessed January 2021.

[4] UTXO-based Blockchain implementations such as Bitcoin allow sophisticated unlocking conditions through their scripting language. Although most people would not call these locking scripts a smart contract, they achieve similar goals in terms of the Blockchain's custodial capabilities.

[5] Source: coinmarketcap.com, accessed September 15th, 2019.

[6] In practice, the collateralization must be much larger, as any credit position with collateralization below 150% is liquidated.

[7] For example, such a token was created in regard to the outcome of the recent U.S. presidential election

REFERENCES

Adams, Hayden. "UniSwap." 2018. https://hackmd.io/@Uniswap/HJ9jLsfTz

Berentsen, Aleksander and Schär, Fabian. "A Short Introduction to the World of Cryptocurrencies" 2018. Vol. 100, Issue 1, pp. 1-16, 2018.

Berentsen, Aleksander and Schär, Fabian. "Stablecoins: The Quest for a Low-Volatility Cryptocurrency." 2019. In: The Economics of Fintech and Digital Currencies. London, pp. 65-71.

Boado, Ernesto. "Aave Protocol Whitepaper v1.0." 2020. https://github.com/aave/aave-protocol/blob/master/docs/Aave_Protocol_Whitepaper_v1_0.pdf

Brooks, Samuel, Jurisevic, Anton, Spain, Michael and Wawrick, Kain. "Havven: a decentralised payment network and stablecoin." 2018. https://www.synthetix.io/uploads/havven_whitepaper.pdf

Buterin, Vitalik. "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform." 2013. http://ethereum.org/ethereum.html.

Cronje, Andre. "Introduction to Yearn." 2020. https://docs.yearn.finance/.

Entrinken, William, Shirley, Dieter, Evans, Jacob and Sachs, Nastassia. "ERC-721 Non-Fungible Token Standard." 2018. https://github.com/ethereum/EIPs/blob/master/EIPS/eip-721.md

Feng, Felix and Weickmann, Brian. "Set: A Protocol for Baskets of Tokenized Assets (v1.2)." 2019. https://www.setprotocol.com/pdf/set_protocol_whitepaper.pdf

Hertzog, Eyal, Benartzi, Guy and Benartzi, Galia. "Bancor protocol - continuous liquidity and asynchronous price discovery for tokens through their smart contracts; aka *smart tokens*." 2017. https://whitepaper.io/document/52/bancor-whitepaper

Juliano, Antonio. "dYdX: A Standard for Decentralized Margin Trading and Derivatives." 2017. https://whitepaper.dydx.exchange/

Leshner, Robert and Hayes, Geoffrey. "Compound: The Money Market Protocol." 2019. https://compound.finance/documents/Compound.Whitepaper.pdf

Liu, Zebang and Palayer, Guillaume. "Betoken: A Meritocratic Hedge Fund Built on Ethereum." 2018. https://github.com/Betoken/Whitepaper/blob/master/BetokenWhitepaper.pdf

Luu, Loi and Velner, Yaron. "KyberNetwork: A trustless decentralized exchange and payment service." 2017. https://whitepaper.io/document/43/kyber-network-whitepaper

Oved, Michael and Mosites, Don. "Swap: A Peer-to-Peer Protocol for Trading Ethereum Tokens." 2017. https://swap.tech/pdfs/SwapWhitepaper.pdf

Peterson, Jack, Krug, Joseph, Zoltu, Micah, Williams, Austin K. and Alexander, Stephanie. "Augur: A Decentralized Oracle and Prediction Market Platform (v2.0)." 2019. https://www.augur.net/whitepaper.pdf

Roth, Jakob and Schär, Fabian and Schöpfer, Aljoscha. "The Tokenization of Assets: Using Blockchains for Equity Crowdfunding." 2019. http://dx.doi.org/10.2139/ssrn.3443382

Szabo, Nick. "Smart Contracts." 1994. http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html

Szabo, Nick. "The idea of smart contracts." 1997. https://nakamotoinstitute.org/the-idea-of-smart-contracts/

Trinkler, Reto and El Isa, Mona. "Melon protocol: a blockchain protocol for digital asset management." 2017. https://github.com/melonproject/paper/blob/master/melonprotocol.pdf

Vogelsteller, Fabian and Buterin, Vitalik. "ERC-20 Token Standard." 2015. https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md

Warren, Will and Bandeali, Amir. "0x: An open protocol for decentralized exchange on the Ethereum blockchain." 2017. https://0x.org/pdfs/0x_white_paper.pdf

Wolff, Max. "Introducing Marble: A Smart Contract Bank." 2018. https://medium.com/marbleorg/introducing-marble-a-smart-contract-bank-c9c438a12890

Wood, Gavin. "Ethereum: A Secure Decentralised Generalised Transaction Ledger." 2015. https://ethereum.github.io/yellowpaper/paper.pdf.

Zhang, Yi, Chen, Xiaohong and Park, Daejun. "Formal specification of constant product (x times y = k) market maker model and implementation." 2018. https://github.com/runtimeverification/verified-smart-contracts/blob/uniswap/uniswap/x-y-k.pdf

ACKNOWLEDGEMENTS

The author would like to thank two anonymous reviewers for their valuable comments. Special thanks go to Florian Bitterli, Raphael Knechtli and Tobias Wagner for their support with data collection and visualization and to Emma Littlejohn and Amadeo Brands for proof-reading.