Ocean Protocol:
Tools for the Web3 Data Economy

Ocean Protocol Foundation[1] with BigchainDB GmbH[2]

Technical Whitepaper, Version 2020-DEC-09

Abstract

Ocean Protocol is an on-ramp for data services into crypto ecosystems, using *datatokens*. Each datatoken is a fungible ERC20 token to access a given data service. Ocean smart contracts and libraries make it easy to publish data services (deploy and mint datatokens) and consume data services (spend datatokens). Ocean contracts run on Ethereum mainnet to start, with other deployments to follow. Ethereum composability enables crypto wallets as data wallets, crypto exchanges as data marketplaces, data DAOs as data co-ops, and more.

Ocean Market is an open-source community marketplace for data. It supports automatic determination of price using an "automated market maker" (AMM). Each datatoken has its own AMM pool, deployed in a gas-efficient manner. Anyone can add liquidity, aka *stake* (equivalent in AMMs). This is *curation*, as stake is a proxy to dataset quality. Publishing amounts to an "initial data offering" (IDO).

We envision thousands of data marketplaces, where Ocean Market is just one. In addition to Ocean Market being open-source (and therefore forkable), Ocean includes tools to help developers build their own marketplaces and other apps.

Ocean's "Compute-to-Data" feature gives compute access on *privately held* data, which never leaves the data owner's premises. Ocean-based marketplaces enable monetization of private data while preserving privacy.

These tools are part of a system designed for long-term growth of a permissionless Web3 Data Economy. The Ocean Data Farming program incentivizes a supply of data. The community-driven OceanDAO funds software development, outreach, and more. Its income is Ocean apps revenue, Ocean Network Rewards (51% of OCEAN supply) and Ocean Protocol Foundation treasury.

Token dynamics are designed such that OCEAN health rises with usage volume.

[1] oceanprotocol.com
[2] bigchaindb.com

Contents

1. Overview

1.1. Introduction

Modern society runs on data [Economist2017]. Modern artificial intelligence (AI) extracts value from data. More data means more accurate AI models [Banko2001] [Halevy2009], which in turn means more benefits to society and business. The greatest beneficiaries are companies that have *both* vast data and internal AI expertise, like Google and Facebook. In contrast, AI startups have excellent AI expertise but are starving for data; and typical enterprises are drowning in data but have less AI expertise. The power of both data and AI—and therefore society—is in the hands of few.

The aim of Ocean Protocol is to spread the benefits of AI by equalizing the opportunity to access and monetize data. We accomplish this by creating simple *tools* to publish data and consume data as decentralized *datatokens*. Datatokens interoperate with ERC20 wallets, exchanges, DAOs and more. This data may be held on-premise to preserve privacy. Additionally, Ocean has tools for data marketplaces. We have implemented these tools as Solidity code running on Ethereum mainnet [Buterin2013]; as Python and JavaScript/React libraries to ease higher-level integration; and as a community data marketplace web application. Over time, Ocean will also get deployed to other networks.

These tools are encapsulated in a broader system design for long-term growth of an open, permissionless data economy. The Data Farming program incentivizes a supply of data. OceanDAO funds software development, outreach, and more. OceanDAO will be funded by revenue from apps and services in the Ocean data ecosystem, Ocean network rewards, and Ocean Protocol Foundation. These are all detailed in the paper.

Everything described above has been deployed live as of Nov 30, 2020.

1.2. Paper Organization

The rest of this paper is organized as follows.

Ocean System. Section 2 describes Ocean's system-level design, which is designed to facilitate the emergence of an open data ecosystem.

Ocean Subblocks. Then, several sections describe key building blocks in the Ocean system:

- Section 3 describes Ocean Tools to power a Data Ecosystem, including datatokens, tools software architecture, network deployment, data marketplaces, and Compute-to-Data.
- Section 4 describes OceanDAO.
- Section 5 describes Network Rewards.

Ocean Data Farming. Section 6 describes a program to kickstart a supply of quality data.

OCEAN token. Section 7 describes the OCEAN token model, bringing together information presented in prior sections.

Applications. Section 8 describes applications of Ocean.

Final. Section 8 concludes. Section 9 holds references.

2. Top-Level Block: Ocean System

2.1. Goals

The top-level goal is to spread the benefits of AI by equalizing the opportunity to access and monetize data. We can make this more specific. The Ocean System has these aims:

- An overall system that is **sustainable and growing**, towards ubiquity.
- Basic design is **simple** to understand and communicate.
- In line with **Ocean Mission and Values** [McConaghy2018]. These include: unlock data, preserve privacy, spread of power, spread of wealth, stay within the law, censorship-resistant & trustless. It should be permissionless, rent-free, and useful to the world [Goldin2017]. It should be anti-fragile: get more resilient when "kicked", therefore also needing evolvability. It follows time scales of decades not months.

2.2. The Design

Figure 1 shows the Ocean System design. At its heart is a **loop**, designed for "snowball effect" growth of the ecosystem. The **Workers** (center) do ***work*** to help grow the **Data Ecosystem** (right). Marketplaces and other data ecosystem services generate revenue, using Ocean software tools. A tiny fraction of that revenue is **looped back** (arrow looping from right to left) as **Network revenue** to the Ocean community: to **Buy & Burn OCEAN** (bottom left) and back to workers curated by **OceanDAO** (center-left). To catalyze growth and ensure decent funding in early days, **Network rewards** (left) also feed to Workers via OceanDAO.

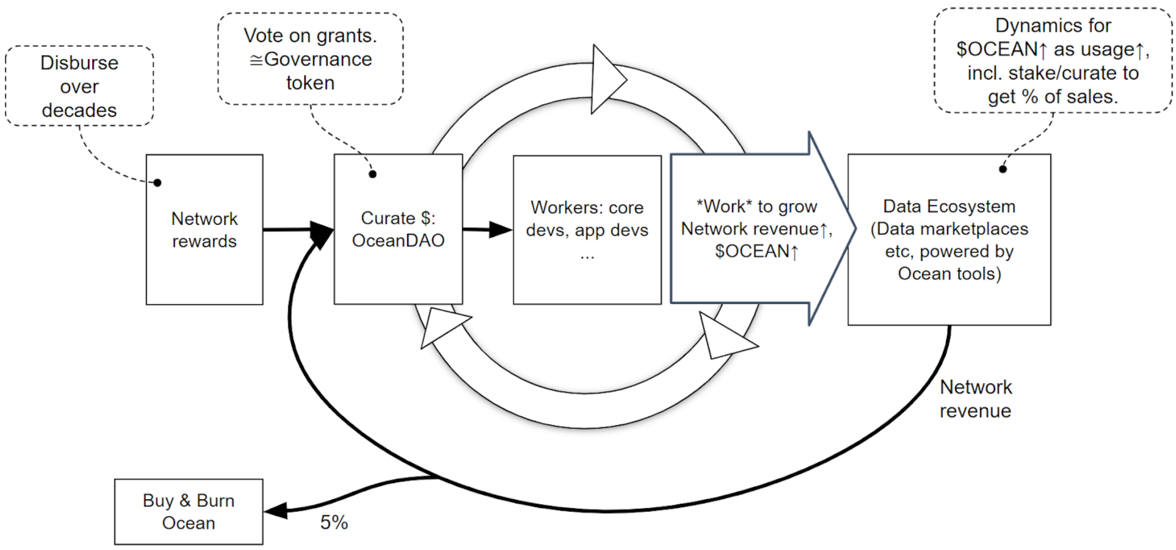

Figure 1: Ocean System

2.3. Kickstarting Growth, and Long-Term Sustainability

The system design achieves sustainability over decades via funding that goes to teams (Workers) that continually add value to the Data Ecosystem.

Funding comes from Network Revenue and from Network Rewards. In the longer term, the bulk of revenue will be from Network Revenue. To help kickstart growth, **51%** of the overall OCEAN token

supply is dedicated to Network Rewards, which follow a Bitcoin-like disbursement schedule (but with a ramp-up period).

The Ocean token is designed such that as usage of Ocean grows, it grows the value accrued in OCEAN. Here's the loop: more usage of Ocean tools (in the data ecosystem) leads to more Network Revenue, which goes to burning and OceanDAO. Burning OCEAN reduces supply, to grow the value accrued in OCEAN. Funds go through OceanDAO to workers who have the mandate to grow usage of Ocean tools. And the loop repeats.

OceanDAO curates funding coming from Network Revenue and Network Rewards. It gets people to do "work" such as improve Ocean core software, build applications using Ocean, spread awareness, grow data supply, and more. A Data Farming program incentivizes data supply.

This system design generalizes beyond Ocean; we can call it the "Web3 Sustainability Loop". It draws inspiration from businesses and nations. The essay [McConaghy2020d] elaborates.

3. Subblock: Data Ecosystem Powered by Ocean Tools

This section and the ones that follow add details on subblocks of the Ocean system.

3.1. Introduction

The Data Ecosystem is a major subblock of the Ocean system. **Ocean tools** power the Data Ecosystem. This section describes Ocean tools.

3.2. USPs of Ocean Tools

Towards *building something that people want* [Graham2005], Ocean tools offer these unique selling propositions (USPs):

- **Earn by selling data, and staking on data.** Ocean Market makes it easy to sell your data, whether you are an individual, company, or city. Furthermore, anyone can stake on data to earn a % of transaction fees.
- **An on-ramp and off-ramp for data assets into crypto**, allowing: crypto wallets for data custody & data management, DEXes[3] for data exchanges, DAOs[4] for data co-ops, securitizing data assets, and more via DeFi composability. It's "data legos." The data itself does not need to be on-chain, just the access control.
- **Quickly launch a data marketplace, with many USPs**: buy & sell private data while preserving privacy, non-custodial, censorship-resistant, auto price discovery, data audit trails, and more.
- **Decentralized data exchange platform**, enabling these characteristics: improve the visibility, transparency and flexibility in usage of data; share data while avoiding "data escapes"; needs little dev-ops support and maintenance; has high liveness; is non-custodial; and is censorship-resistant. E.g. Ocean as a traffic data management platform for smart cities. E.g. federated learning without having to trust the orchestration middleman [McConaghy2020c].

3.3. Ocean Tools Foundation: Datatokens

3.3.1. Datatokens Introduction

Ocean is an on-ramp and off-ramp for data assets into crypto ecosystems, using *datatokens*. Datatokens are ERC20 tokens [Vogelsteller2015] to *access* data services. Each data service gets its

[3] DEX = Decentralized Exchange
[4] DAO = Decentralized Autonomous Organization

own datatoken. Ocean smart contracts and libraries make it easy to publish data services (deploy and mint datatokens) and consume data services (spend datatokens).

3.3.2. Datatokens Goals

Here are the main goals.

- **Simple.** Complexity is *the* challenge in software design. Accordingly, we strive to make Ocean's overall design simple. This includes **simple developer experience**, **simple user experience**, and **simple code** (sufficiently simple to deploy to Ethereum mainnet, for additional benefits of being permissionless, stability, security, composability, and community).
- **Be a tool for existing workflows**; focus where Ocean adds value. "Tornado Cash and Uniswap … are successful in part because they are just tools that people can put into their existing workflows, and not ecosystems" [Buterin2020]. We agree. Platforms imply asking you to *switch* from another platform. In contrast, tools humbly ask you to try them out, and if you find them useful, to add them to your toolbox. Platforms are zero-sum; tools are positive-sum. This means Ocean should **leverage other infrastructure** wherever possible, and be maximally composable with other protocols and tools.

Ocean datatokens achieve these goals. They keep Ocean simple, composable, and make Ocean more a set of tools and less a platform.

3.3.3. Datatokens Mental Model

Figure 2 shows the mental model for Ocean datatokens. Ocean does the beginning (create datatokens) and the end (consume datatokens). In between are any ERC20-based applications, including Ocean-based marketplaces.

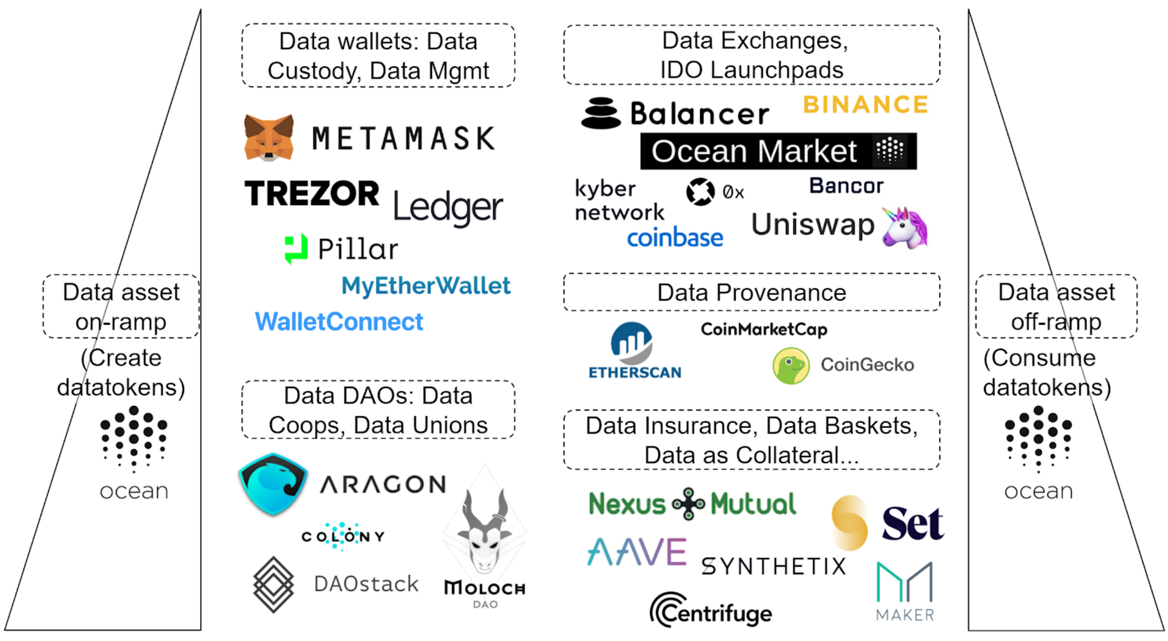

Figure 2: Mental model. Datatokens are the interface to connect data assets with DeFi tools. Ocean is an on-ramp for data services into ERC20 datatoken data assets on Ethereum, and an off-ramp to consume data. [Note: showing a logo does not imply a partnership]

3.3.4. Datatokens are ERC20 Access Tokens

Traditional access tokens exist, such as OAuth 2.0 [Oauth2020]. If you present the token, you can get access to the service. However, the “tokens” are simply a string of characters, and “transfer” is basically copying and pasting that string. This means they can easily be “double-spent”: if one person gets access, they can share that access with innumerable others, even if that access was only meant for them. These tokens aren’t the “tokens” we think of in blockchain.

How do we address the double-spend problem? This is where blockchain technology comes in. In short, there’s a single shared global database that keeps track of who owns what, and can then easily prevent people from spending the same token twice. Footnote ([5]) gives details.

This generalizes beyond Bitcoin tokens to other token assets. ERC20 [Vogelsteller2015] was developed as a standard for token ownership actions. It’s been adopted widely in Ethereum and beyond. Its focus is *fungible* tokens, where tokens are fully interchangeable.

We can connect the idea of *access* with the ERC20 token standard. Specifically, consider an ERC20 token where **you can access the dataset if you hold 1.0 tokens**. To access the dataset, you send 1.0 datatokens to the data provider. You have custody of the data if you have at least 1.0 tokens. To give access to someone else, send them 1.0 datatokens. That’s it! But now, the double-spend problem is solved for “access control”, and by following a standard, there’s a whole ecosystem around it to support that standard.

Datatokens are ERC20 tokens to access data services[6]. Each data service gets its own datatoken.

3.3.5. Datatoken Variants

At the smart contract level, datatokens don’t differ. Variants emerge in the semantic interpretation by libraries run by the data provider, one level up. Here are some variants:

- Access could be **perpetual** (access as many times as you like), **time-bound** (e.g. access for just one day, or within specific date range), or **one-time** (after you access, the token is burned).

[5] Let’s illustrate how the Bitcoin system prevents double-spending of Bitcoin tokens (bitcoin). In the Bitcoin system, you “control” an “address”. An “address” is a place where bitcoin can be stored. You “control” the address if you’re able to send bitcoin from that address to other addresses. You’re able to do that if you hold the “private key” to that address. A private key is like a password -- a string of text you keep hidden. In sending bitcoin, you’re getting software to create a transaction (a message) that specifies how much bitcoin is being sent, and what address it’s being sent to. You demonstrate it was you who created the transaction, by digitally signing the message with your private key associated with your address. The system records all such transactions on this single shared global database with thousands of copies shared worldwide.

[6] We could also use ERC721 “non-fungible tokens” (NFTs) [ERC721] for data access control, where you can access the dataset if you hold the token. Each data asset is its own "unique snowflake". However, datasets typically get shared among >1 people. For this we need fungibility, which is the realm of ERC20.
However there is a more natural fit for NFTs: use an NFT to represent the base rights. The base rights are the ability to create access licenses (=mint ERC20 access tokens). The first base rights holder is the copyright holder, but they could transfer this to another entity as an exclusive deal (= transfer the NFT).
Ocean's V3 datatokens release uses just ERC20 tokens. There is a base rightsholder, it's just implicit: it's the entity that controls the ability to mint more ERC20 tokens (= "publisher"). This could conceptually be replaced with an explicit representation, which is NFT.

- Data access is always treated as a data service. This could be a service to access a **static** dataset (e.g. a single file), a **dynamic** dataset (stream), or for a compute service (e.g. "**bring compute to the data**"). For static data, we can tune variants based on the type of storage: Web2 cloud (e.g. AWS S3), Web3 non-permanent (e.g. Filecoin), Web3 permanent small-scale (e.g. Ethereum), Web3 permanent large-scale (e.g. Arweave), or go meta using IPFS but "pinned" (served up) by many places. For dynamic data, variants include Web2 streaming APIs (single-source), Web3 public data oracles (e.g. Chainlink), and Web3 private data oracles (e.g. DECO).
- **Read vs write etc access.** This paper focuses on "read" access permissions. But there are variants: Unix-style (read, write, execute; for individual, group, all); database-style (CRUD: create, read, update, delete [CRUD2020]), or blockchain database-style (CRAB: create, read, append, burn [Pregelj2017]).

The terms of access are specified in the metadata, which is on-chain (more on this later).

3.3.6. Datatokens and Rights

Having a token to physically access data implies the **right** to access the data. We can formalize this right: the datatoken would typically automatically have a **license** to use that data. Specifically: the data would be **copyrighted** (a form of intellectual property, or IP), as a manifestation of bits on a physical storage device. The license is a contract to use the IP in specific form. In most jurisdictions, copyright happens automatically on creation of the IP. Alternatively, encrypted data or data behind a firewall can be considered as a **trade secret**. Overall, enforcement in a given jurisdiction then falls under its existing IP framework[7].

"Ownership" is a bundle of rights. "Owning" a token means you hold the private key to a token, which gives you the right to transfer that token to others. Andreas Antonopoulos has a saying: "Your keys, your Bitcoin. Not your keys, not your Bitcoin" [Ogundeji2016]. That is, to truly own your Bitcoin, you need to have the keys to it. This crosses over to data: **"Your keys, your data. Not your keys, not your data"**. That is, to truly own your data, you need to have the keys to it.

3.3.7. Relation to Oracles

Oracles like Chainlink help get *data itself* on-chain [Chainlink2020]. Ocean is complementary, providing tools to on-ramp and off-ramp *data assets*. The data itself does not need to be on-chain, which allows wider opportunity for leveraging data in DeFi. Oracle datafeeds can be tokenized using Ocean.

3.3.8. Analogy to Shipping Containers

Here's an "intuition pump" to understand datatokens. Just as shipping containers are an *overlay protocol* that made physical supply chains more efficient, datatokens are an *overlay protocol* that makes data service supply chains more efficient. The essay [McConaghy2020e] elaborates.

[7] The World Intellectual Property Office (WIPO) is a United Nations (UN) agency dedicated to setting IP guidelines. Each jurisdiction can then choose how to implement the guidelines. For example, the United States Patent & Trademark Office (USPTO) implements WIPO guidelines on patents and trademarks in the US.

3.4. Ocean Tools Architecture

3.4.1. Overview

Figure 3 shows the Ocean tools architecture. The lowest level has the smart contracts, which are deployed on Ethereum mainnet[8]. Above that are libraries and middleware, which expose the contracts to higher-level languages and provide convenience utilities. The top layer is applications.

Left to right are groupings of functionality: tools for datatokens, tools for markets (including pools), tools to consume data services and for metadata, and external ERC20 tools. The following subsections elaborate; more information yet is in the documentation [OceanDocs2020] and the open-source code itself [OceanContracts2020] [OceanLibPy2020] [OceanLibJs2020] [OceanReact2020] [OceanMarket2020].

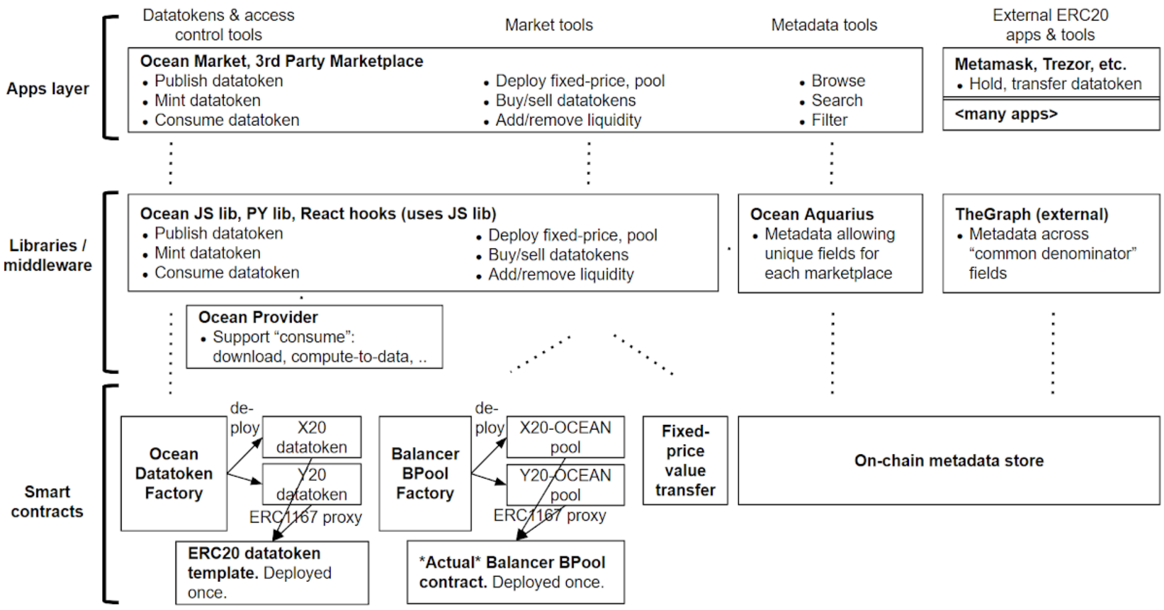

Figure 3: Ocean tools architecture

3.4.2. Datatokens & Access Control Tools

The publisher actor holds the dataset in Google Drive, Dropbox, AWS S3, on their phone, on their home server, etc. The dataset has a URL. The publisher can optionally use IPFS for a content-addressable URL. Or instead of a file, the publisher may run a compute-to-data service.

In the **publish** step, the publisher invokes **Ocean Datatoken Factory** to deploy a new datatoken to the chain. To save gas fees, it uses ERC1167 proxy approach on the **ERC20 datatoken template**. The publisher then mints datatokens.

The publisher runs **Ocean Provider**. In the **consume** step, Provider software needs to retrieve the data service URL given a datatoken address. One approach would be for the publisher to run a database; however this adds another dependency. To avoid this, it stores the URL on-chain. So that others don't see that URL, it encrypts it.

To initiate the **consume** step, the data consumer sends 1.0 datatokens to the Provider wallet. Then they make a service request to the Provider. The Provider loads the encrypted URL, decrypts it, and

[8] Deployed to Ethereum mainnet to start, then to other networks in time. A later section elaborates.

provisions the requested service (send static data, or enable a compute-to-data job). The Appendix has details of this process.

Instead of running a Provider themselves, the publisher can have a 3rd party like Ocean Market run it. While more convenient, it means that the 3rd party has custody of the private encryption/decryption key (more centralized). Ocean will support more service types and url custody options in the future.

Ocean JavaScript and Python libraries act as drivers for the lower-level contracts. Each library integrates with Ocean Provider to provision & consume data services, and Ocean Aquarius for metadata. **Ocean React hooks** use the JavaScript library, to help build webapps & React Native apps with Ocean.

3.4.3. Market Tools

Once someone has generated datatokens, they can be used in any ERC20 exchange, including AMMs. We elaborate on this later. In addition, Ocean provides **Ocean Market**. It's a vendor-neutral reference data marketplace for use by the Ocean community. It's decentralized (no single owner or controller), and non-custodial (only the data owner holds the keys for the datatokens).

Ocean Market supports fixed pricing and automatic price discovery. For fixed pricing, there's a simple contract for users to buy/sell datatokens for OCEAN, while avoiding custodianship during value transfer.

For automatic price discovery, Ocean Market uses Balancer pools [Balancer2020a]. Each pool is a datatoken - OCEAN pair. In the Ocean Market GUI, the user adds liquidity then invokes pool creation; the GUI's React code calls the Ocean JavaScript library, which calls **Balancer Factory** to deploy a **Balancer BPool** contract. (The Python library also does this.) Deploying a datatoken pool can be viewed as an "Initial Data Offering" (IDO).

Complementary to Ocean Market, Ocean has reference code to ease building **third-party data marketplaces**, such as for logistics (dexFreight data marketplace [OceanBlog2020a]) or mobility (Daimler [OceanBlog2020b]).

3.4.4. Metadata Tools

Metadata (name of dataset, dateCreated etc.) is used by marketplaces for data asset discovery. Each data asset can have a decentralized identifier (DID) [DID2019] that resolves to a DID document (DDO) for associated metadata [OEP7_2019]. The DDO is essentially JSON [JSON2020] filling in metadata fields.

OEP8 [OEP8_2019] specifies the metadata schema, including fields that must be filled. It's based on the public DataSet schema from schema.org [SchemaOrg2020].

Ocean uses the Ethereum mainnet as an **on-chain metadata store**, i.e. to store both DID and DDO. This means that once the write fee is paid, there are no further expenses or dev-ops work needed to ensure metadata availability into the future, aiding in the discoverability of data assets. It also simplifies integration with the rest of the Ocean system, which is Ethereum-based. Storage cost on Ethereum mainnet is not negligible, but not prohibitive and the other benefits are currently worth the tradeoff compared to alternatives.

Due to the permissionless, decentralized nature of data on Ethereum mainnet, any last-mile tool can access metadata. **Ocean Aquarius** supports different metadata fields for each different Ocean-based marketplace. Third-party tool **TheGraph** sees metadata fields that are common across all marketplaces.

3.4.5. Third-Party ERC20 Apps & Tools

The ERC20 nature of datatokens eases composability with other Ethereum tools and apps, including **MetaMask** and **Trezor** as data wallets, DEXes as data exchanges, and more. The Applications section expands on this.

3.4.6. Actor Identities

Actors like data providers and consumers have Ethereum addresses, aka web3 accounts. These are managed by crypto wallets, as one would expect. For most use cases, this is all that's needed. There are cases where the Ocean community could layer on protocols like Verifiable Credentials [W3C2019] or tools like 3Box [3Box2020].

This subsection has described the Ocean tools architecture at a higher level. [McConaghy2020h] [OceanDocs2020] has further details.

3.5. Ocean Tools: Network Deployments

Towards a broadly open Data Economy, we aim for Ocean deployment across many chains as a thin layer for data assets and permissioning. It starts with a single deployment, and expands.

3.5.1. Initial Permissioned Deployment

"Decentralized" means no single point of failure. "Permissioned" means there are a set of gatekeepers that together control the entity. "Permissionless" means there are no gatekeepers; one needs no permission to have a hand in controlling an entity.

Ocean V1 and V2 have been decentralized and *permissioned*:

- Ocean smart contracts run on a permissioned Proof-of-Authority (POA) network [McConaghy2019a].
- A small group of people can upgrade the smart contracts (in a multisig setting).

3.5.2. Permissionless via Ethereum Mainnet

Starting with Ocean V3.0 release, Ocean is decentralized and *permissionless*.

- **Ocean smart contracts are deployed to Ethereum mainnet** - a permissionless network.
- **The contracts will not have upgradeability built in**. Therefore the only way to upgrade contracts is by *community consensus* to use a new set of smart contracts. (There's no longer a small handful of gatekeepers.)

This means that Ocean V3 also meets the "V5" target of being permissionless, specified in the Ocean roadmap [OceanBlog_Roadmap2019] [OceanBlog_Roadmap2020].

Despite using Ethereum mainnet, the "scale" aspect is now manageable [McConaghy2016]. This is possible because Ocean V3 contracts are simpler than the V1 and V2 contracts due to the datatokens architecture and more. This includes greatly reduced gas usage, and fewer wallet confirmations to purchase data. Finally, we have realized that data scientists are accustomed to some latency with Web2 payments; we don't need to be radically better than that.

3.5.3. Reduced Gas Costs via Sister Chain

The growth of DeFi has meant increased usage of Ethereum mainnet, and in turn a great increase in gas prices. To alleviate this cost, and for improved scalability, Ocean V3.x will have a “sister” deployment to another network. It may be Ocean V3.1 depending on other priorities. One possibility is xDai Chain [Xdai2020a], with an Arbitrary Message Bridge (AMB) [TokenBridge2020] to connect OCEAN tokens *and* datatokens to Ethereum mainnet. Other strong possibilities include a Substrate + EVM deployment [Parity2020], or a rollup-based technology like OVM [EthOptimism2020].

It's acceptable if the sister chain has a lower security, since users would have the choice between higher security (Ethereum mainnet) or lower gas costs (the sister, at potentially lower security). Then, data assets that start high value or need high security can continue to deploy directly to Ethereum mainnet. And, lower-value data assets can first get deployed to the sister network, then "graduate" to Ethereum mainnet if they increase sufficiently in value[9].

3.5.4. Further Network Deployments

Over time, we envision every blockchain network to have an Ocean-powered data resource permissioning layer. We envision them being interconnected with datatokens and OCEAN flowing everywhere.

We envision Ocean as an add-on library in Parity Substrate [Parity2020] and [CosmosSdk2020] Cosmos SDK, making it only an "import" away for Polkadot and Cosmos blockchain families respectively. Substrate 2.0 support for off-chain data integration is particularly promising.

We see potential integrations with layer 2 rollups such as Optimistic Virtual Machine (OVM) to improve privacy, throughput or cost [EthOptimism2020].

We expect to see deployment to other EVM-based networks such as Binance Smart Chain, Matic Network, SKALE Network, NEAR Protocol, and Solana; each with their respective bridges. We envision deployment onto federated networks for consortia like Energy Web Chain. Finally, we eventually see deployment to non-EVM blockchains, especially ones that can hold a lot of data like arweave, or ones with built-in oracles like aeternity.

In all these deployments, OCEAN tokens will remain on Ethereum mainnet, and bridged to other networks. We will be encouraging the broader community to do each of these deployments, with funding from OceanDAO or the respective chain's grant mechanisms. In each of these deployments, with the interest of Ocean sustainability, there will be small transaction fees that go to the Ocean community via a bridge.

3.6. Ocean Tools: Data Marketplaces

3.6.1. Introduction

This section drills into details about Ocean-based data marketplaces.

Since each data service has its own ERC20 token, any ERC20 exchange can serve as a data marketplace. They can be AMM DEXes, order-book DEXes, order-book CEXes, and more.

---

[9] Thanks to Simon de la Rouviere for this framing, which he in turn derived from Austin Griffith.

But we can still make it easier for users. Specifically, marketplaces *tuned* for data can help users in the whole data flow, including publish data, set price, curate data, discover data, buy data, and consume data.

3.6.2. Marketplaces Architecture

Figure 4 shows the conceptual architecture. There are many data marketplace frontends; there is a common backend (for a given network).

The frontends include a community market (**Ocean Market**) [top middle] and independent third-party markets like those of dexFreight or Daimler [top left]. Each frontend runs client-side in the browser, using **Ocean React hooks**, which use the **Ocean JS library**, which interfaces with the backend.

The **Decentralized Backend** [bottom left] is Solidity code running on Ethereum mainnet, and includes the datatoken and pool contracts, and the on-chain metadata store.

When a **Buyer** purchases a dataset on a frontend [far top, far left], most of the revenue goes to the **Data seller** [bottom right]. Some fees go to **LPs, the marketplace runner, and the broader community** [top right].

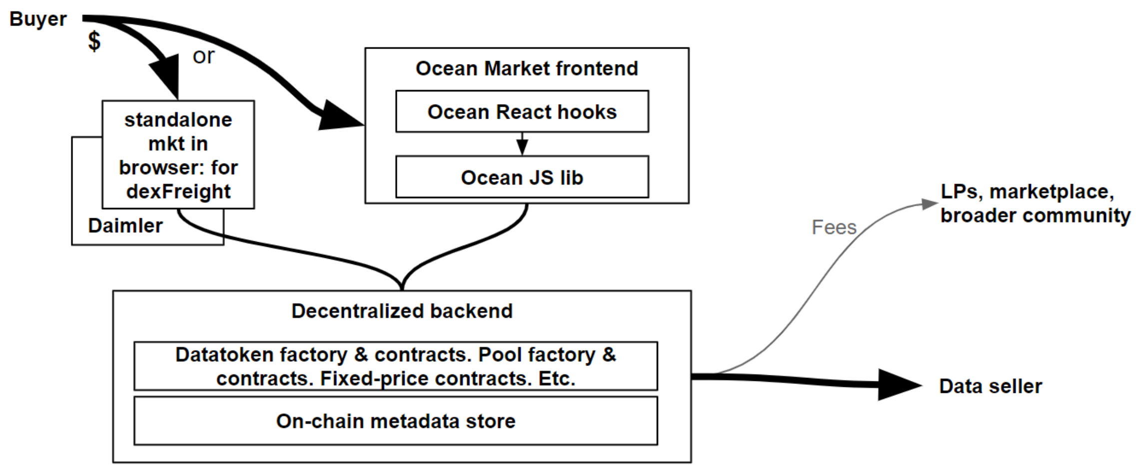

Figure 4: Ocean Marketplaces share a decentralized backend

3.6.3. How Do You Price Data?

"How do I price the data?" is an oft-asked question, and rightly so. It's a real problem. Pricing data is hard [Kuhn2019a] [Kuhn2019b].

3.6.4. Fixed Pricing

Some data sellers will want to sell data for a fixed price (fixed # OCEAN). The challenge is to do it without intermediaries or complicated escrow contracts. Our solution is a simple smart contract that includes *transfer()* OCEAN one way, and *transfer()* datatoken the other way.

3.6.5. Automated Pricing via AMMs

If price can be discovered automatically, it would be of immense value. It's worth spending real effort on how to price data automatically. Order books, auctions, and AMMs are some possibilities. Let's review each.

- For a sale to occur, **order books** require bids and asks to match up in real time, aka a "double coincidence of wants". This is not feasible for newly created long tail assets like datatokens. (However they are useful once a datatoken gets enough liquidity and traders.)
- **Auctions** occur over a time interval, such as an hour or a day. Auctions are useful for an initial pricing, but after that we still want automated price discovery for the rest of the lifetime of the assets. Auctions do not provide this.
- **AMMs** provide automated price discovery without the disadvantages of order books or auctions listed above. AMMs work for an initial asset offering and throughout the asset's lifetime. AMMs don't require a double coincidence of wants; they can be thought of as robots that are always ready to buy or sell.

Ocean's datatoken framing enables people to build data exchanges using any of the above approaches - order books, auctions, or AMMs. That said, it's worth spending effort to make the most promising approach easy to use.

From the analysis above, the most promising approach is *AMMs*. So, we focus on them. A given AMM pool would have (1) the specific datatoken, and (2) some other more-established token such as ETH, DAI, or OCEAN. Having OCEAN as a convenient default (without forcing its use) helps drive demand for OCEAN, which in turns helps long-term sustainability [2]. In short, we focus on datatoken-OCEAN pools.

AMMs auto-discover price of data. In an Ocean AMM-based data market, a datatoken's price automatically goes up more datatokens are bought (as OCEAN is swapped for more datatokens). It goes down as datatokens are sold (as datatokens are swapped for OCEAN).

AMM options. The burgeoning Ethereum DeFi space offers many high-quality options. Bancor was first. Uniswap has low gas costs. Balancer allows non-equal weights among tokens in the pool (e.g. 90/10 versus 50/50). Many more have emerged recently, along with aggregators. One could build an Ocean-based data market with any of these.

3.6.6. The First Data Market AMM: Balancer

For Ocean Market, we chose to focus on one AMM tool to start: **Balancer**. There were a few reasons for this. First, Balancer allows to add liquidity through a single token, unlike most AMMs. Here's how this helps: when a new datatoken pool is published, at first only the publisher has datatokens. For others to add liquidity, they will need to add just OCEAN tokens. Balancer allows this.

Second, Balancer lightens the liquidity burden for publishers. Balancer uniquely allows non-equal weights among tokens in a pool. This allows a pool with 90% weight to the datatoken and 10% to OCEAN. Compared to a 50/50 pool, a data publisher only needs to provide 1/5 of the OCEAN liquidity for the same initial datatoken price. For the future, Balancer gives the possibility to *dynamically* change weights to bootstrap liquidity [McDonald2020] and mitigate impermanent loss [Balancer2019].

The default deployment of Balancer has high gas costs for deploying pools. We overcame this issue as follows. In Ocean, Balancer factory and pool contracts (BFactory, BPool) are tweaked to use the

ERC1167 proxy pattern to reduce gas costs [ERC1167]. They can be viewed as an early version of Balancer V2[10,11]

Benefits. Here are some benefits for Balancer-Ocean AMM data markets. The first is automatic price discovery, as discussed. Next, if the pool is the first market that this datatoken has been deployed to, then we can consider this the data asset's "**initial data offering**" (IDO) to kickstart liquidity. Finally, the AMM is decentralized and non-custodial. Later in this paper, we elaborate further on IDOs and marketplace benefits.

3.6.7. OCEAN Staking and Data Curation

AMMs require liquidity to be provisioned. Liquidity is the number of datatokens and OCEAN in the pool. Anyone can add liquidity. The higher the liquidity, the lower the slippage (change in price) when there is a purchase.

In providing liquidity, the price signal is authentic [Simler2018] or, equivalently, that the "market" is real.

A liquidity provider (LP) is staking because staking and liquidity provisioning are equivalent in AMMs [Monegro2020].

This means:

OCEAN staking is the act of adding liquidity to a datatoken-OCEAN pool.

Furthermore, an LP is curating since the amount of liquidity is a proxy to quality of the data asset.

This means:

Ocean data curation is the act of adding / removing liquidity in a datatoken-OCEAN pool.

Liquidity providers (LPs) earn a cut of the transaction fee proportional to their stake. Since curators are LPs, it means that curators are incentivized to **curate towards the most valuable datasets** because it will earn them the most fees. It also means that to earn more fees, curators are incentivized to **refer** others to the data pools that they've staked on. In addition, this means that **curation in Ocean has authentic signals** of quality because it requires actual skin-in-the-game in the form of liquidity (stake).

3.6.8. Sustainability & Fees

The Ocean System design described earlier is designed for long-term growth and sustainability of Ocean ecosystem. A key aspect is for the network to collect revenue, which is then used to fund community projects to keep growing the ecosystem.

At the same time, we want each individual marketplace (especially 3rd party marketplace) to be able to garner revenue to drive their individual growth.

Finally, we need to make it worthwhile for people to provide liquidity for data AMMs.

---

[10] The Ocean and Balancer teams have collaborated for years. Part of the collaboration between the Ocean and Balancer teams is discussion around Balancer V2. Ocean plans to use Balancer Lab's deployment by Balancer V2, if not sooner.

[11] The Ocean-tweaked Balancer pools are optimized for low gas cost, but because they do not come from Balancer's official factory, they aren't currently eligible for Balancer Liquidity Mining [Martinelli2020]. If users choose, they can deploy datatoken pools using Balancer's official factory, which is more expensive but does receive BAL rewards. Balancer V2 will optimize gas costs, at which point we expect Ocean libraries to use Balancer official factory (and therefore be eligible for BAL rewards).

Description of Fees. The following design meets the challenges. Fees go to three groups: Ocean Community, Marketplace (consume side), and Liquidity Providers (LPs).

- **Ocean Community.** These fees go to OceanDAO for teams to improve the tools, build apps, do outreach, and more. A small fraction also goes to burning OCEAN. These fees are collected in the action of consuming a data asset.
- **Marketplace (Consume side).** This fee goes to the marketplace that was present for consume-side functionality. In Ocean Market, these fees are collected in the action of consuming a data asset [6] and are treated as Network Revenue that goes to the Ocean Community. Marketplaces can generate additional revenue by selling data themselves, being LPs, and (with more effort) taking fees during the buy/sell act. Taken together, these fees help towards true data marketplace *businesses* that can grow sustainably over time.
- **Liquidity Providers (LPs)** take corresponding transaction fees on datatoken purchases. The fee is set by the creator of the pool. Ocean provides a default of 0.1%. LPs are incentivized to *refer* others to their pools to earn more fees. Therefore AMMs work a bit like decentralized affiliate programs.

The Community fee is hardcoded to 0.1% in the datatoken template. The marketplace consume-side fee is hardcoded to 0.1%. The LP fee is set by the pool creator, which in Ocean Market is typically the Provider. It defaults to 0.1%. Therefore the total fee defaults to 0.1% LP + 0.1% marketplace + 0.1% community = 0.3%. These defaults are low to avoid rent extraction, and are par with Uniswap's 0.3% fee [Uniswap2020].

3.6.9. Ocean Marketplace Actions

Functionality covered. A marketplace's core functionality is about connecting buyers to sellers for given assets: to make the assets discoverable, and and buying/selling an asset of interest. For smoother user flow, Ocean Market supports adjacent functionality: publishing the asset in the first place, and consuming it. Each subsection will cover these, in the order that it would happen, with a focus on Ocean Market.

Action: publish dataset. When the user (publisher) clicks on "Publish", they end up here. They start to fill out metadata, including Title and Description. The publisher then provides the url of where the data asset can be found. This url gets stored encrypted and on-chain. When a buyer later consumes a datatoken, that url will be decrypted. The publisher then fills out price information. It may be fixed price or dynamic (automatic). If automatic, they add liquidity as desired to be in line with their target price. Finally, they hit "publish" and Ocean Market will invoke blockchain transactions to deploy a datatoken contract, publish metadata on-chain, and (if automatic) do all AMM-related transactions.

Action: add/remove liquidity. If the datatoken has an AMM pool, *any* user can add/remove datatokens or OCEAN as liquidity. This is OCEAN staking, and curation, as discussed earlier.

Action: discovery. Ocean Market will have thousands of data assets. To help discovery, Ocean includes support for browsing, searching, and filtering data assets.

Action: buy/sell dataset. Here, a buyer comes to Ocean Market and connects their wallet. Their wallet has some OCEAN. The buyer clicks the "buy" button; then Metamask pops up and asks for the buyer to confirm a transaction to swap OCEAN tokens for 1.0 datatokens.

The buyer confirms, and the swap happens on-chain. Now, the buyer now has 1.0 more datatokens in their wallet.

Action: Consume Dataset. Here, a datatoken owner comes to Ocean Market and connects their wallet. They go to the appropriate sub-page with the asset that they own. They consume the dataset by clicking the "use" button. They follow the prompts to end up with a downloaded dataset, or to get results of bringing compute to data.

3.6.10. Developer Tools & 3rd Party Marketplaces

Ocean Market is just one data marketplace. We envision *many* data marketplaces. We can catalyze this, by making it easy for developers to create their own marketplaces. Using Ocean JavaScript or Python libraries, each of the following is 1–3 lines of JS or PY code:

- Create a data asset (provision data service, deploy datatoken contract, add metadata, mint datatokens)
- Create an AMM market (or fixed-price market)
- Add or remove liquidity
- Swap OCEAN for datatokens, & vice versa (buy datatokens & sell datatokens)
- Submit a datatoken and consume a data asset

The libraries interface to Ocean smart contracts. Ocean Provider is a support tool to provision data assets, and Ocean Aquarius and to help store data on-chain and to query metadata (with the help of a local cache).

With these tools in place, there two main ways that a developer can build an Ocean-based data marketplace: (1) Fork Ocean Market, which uses Ocean tools, and (2) Build up their own marketplace using Ocean tools more directly (React hooks, Javascript library, etc).

3.6.11. Group-Restricted Access in Marketplaces

Certain use cases need to restrict who can consume data. For example, only registered medical personnel can read sensitive patient data. Or, only bank employees can review a consumer's KYC application form.

Places to restrict access include:

1. Restrict inside the ERC20 contract itself.
2. Restrict access to the marketplace.
3. Restrict the ability to buy in the marketplace.
4. Restrict at the point of consumption.

(1) involves modifying the ERC20 contract's transfer() and approve() or transferFrom(). Other projects have taken this route. However this alters the spec of ERC20 contracts, which we are reluctant to do.

(2) typically means the marketplace firewalling itself, and only allowing login with verified accounts. This is fairly common practice in the enterprise, and can work here.

(3) means the marketplace maintaining a whitelist of Ethereum addresses that can purchase a given data asset.

(4) means that the provider runs code that checks the person's credentials. Consumption is only allowed if the token is transferred and the credentials check out. This is akin to entering an R-rated movie: you need the ticket (token) and to show your ID (verifiable credentials).

We recommend (2), (3) and (4), depending on the scenario. (2) and (3) cannot fully prevent bad actors from acquiring a restricted datatoken outside of marketplaces, but they are easier to implement (and Ocean tools have affordances for this). (4) is the most secure but takes more effort by the provider.

There are a few blockchain-compatible ways to implement access at a group level. One is to use Verifiable Credentials (VCs), wherein an issuing authority digitally signs an attestation that a DID has a credential [W3C2019]. Another approach is to do a whitelist, via a Token-Curated Registry (TCR), or a custom smart contract implementation. A final way is to use a DAO with particular membership rules.

The Applications section details some use cases for group-restricted access.

3.6.12. Benefits of Ocean Data Marketplaces

Ocean marketplace tools make it easy to build & launch data marketplaces.

Ocean-based data marketplaces have these characteristics:

- *Interoperability* - data assets being bought and sold are ERC20 tokens on the Ethereum mainnet, which play well with the broader Ethereum ecosystem.
- *Don't need login* - users just connect their Web3 wallet (Metamask, etc). Therefore to buy or sell datatokens, they're in and out in 2 minutes. This big UX improvement feels similar to DEXes, versus traditional CEXes[12].
- *Non-custodial & decentralized* - no centralized middlemen controls the datatokens. No single point of failure.
- *Censorship-resistant, with flexibility* - by default, everyone can transact with the marketplace on the same terms, regardless of their personal identifying characteristics. Or, to meet data regulations or KYC, there is the option of whitelists.
- *Buy & sell private data while preserving privacy* - using Ocean Compute-to-Data. Data won't leave the premises. This also gives sellers the option to make data *exclusive (in an economic sense)* which can give data a pricing premium.
- *Provenance* - sellers and buyers benefit from the auditability of purchase transactions (using e.g. Etherscan).
- *Monetization* - marketplace has the option to take a commission on sales. This helps to ensure that data marketplace businesses can be built that can sustain themselves and grow over time.

AMM pools enable these additional characteristics:

- *Automated price discovery* - the pool holds OCEAN and datatoken as liquidity. Datatoken price goes up as more datatokens are sold.
- *Curation (= Staking = Provisioning liquidity)* - authentic signals for quality of a dataset (= amount of OCEAN staked).
- *Transaction fees for LPs / curators / stakers.*
- *Referrals* - LPs are incentivized to refer to pools that they get transaction fees from.

The essay [McConaghy2020f] elaborates on more aspects of Ocean-powered marketplaces, especially Ocean Market. Also, [McConaghy2020i] and [McConaghy2020j] elaborate on staking and selling data, respectively.

[12] CEX = Centralized Exchange

3.7. Ocean Tools: Compute-to-Data

Ocean Compute-to-Data provides a means to share or monetize one's data while preserving privacy. This section expands on this.

3.7.1. Motivation

Private data is data that people or organizations keep to themselves, or at least *want to* keep to themselves. It can mean any personal, personally identifiable, medical, lifestyle, financial, sensitive or regulated information.

Privacy tools are about asymmetric information sharing: get info to the people you want, for their benefit, while ensuring that others don't see the info.

Benefits of Private Data. Private data can help research, leading to life-altering innovations in science and technology. For example, more data improves the predictive accuracy of modern AI models. Private data is often considered the most valuable data because it's so hard to get, and using it can lead to potentially big payoffs. It's often considered as a competitive, or even decisive, advantage in their market by companies.

Risks of Private Data. Sharing or selling private data comes with risk. What if you don't get hired because of your private medical history? What if you are persecuted for private lifestyle choices? Large organizations that have massive datasets know their data is valuable — and potentially monetizable — but do not pursue the opportunity for risk of data escaping and the related liability.

Resolving the Tradeoff. There appears to be a tradeoff between benefits of using private data, and risks of exposing it. What if there was a way to get the benefits, while minimizing the risks? This is the idea behind Compute-to-Data: let the data stay on-premise, yet allow 3rd parties to run specific compute jobs on it to get useful analytics results like averaging or building an AI model. The analytics results help in science, technology, or business contexts; yet the compute is sufficiently "aggregating" or "anonymizing" that the privacy risk is minimized.

3.7.2. Conceptual Working

Figure 5 illustrates Compute-to-Data conceptually. Alice the data scientist goes to a **data marketplace** and purchases access to private data from seller Bob. She runs her AI **modeling algorithm** (which Bob has approved) on Bob's **private data** to **privately train a model**, which Bob also stores privately. She then runs the trained **private model** on new input data to get **model predictions**. Those predictions are the only data she sees. Everyone is satisfied: Alice gets predictions she wants, and Bob keeps his data private.

At the heart, there are **datatokens** which grant access to run compute next to the data. These datatokens can be used within a marketplace context (like described) or other contexts.

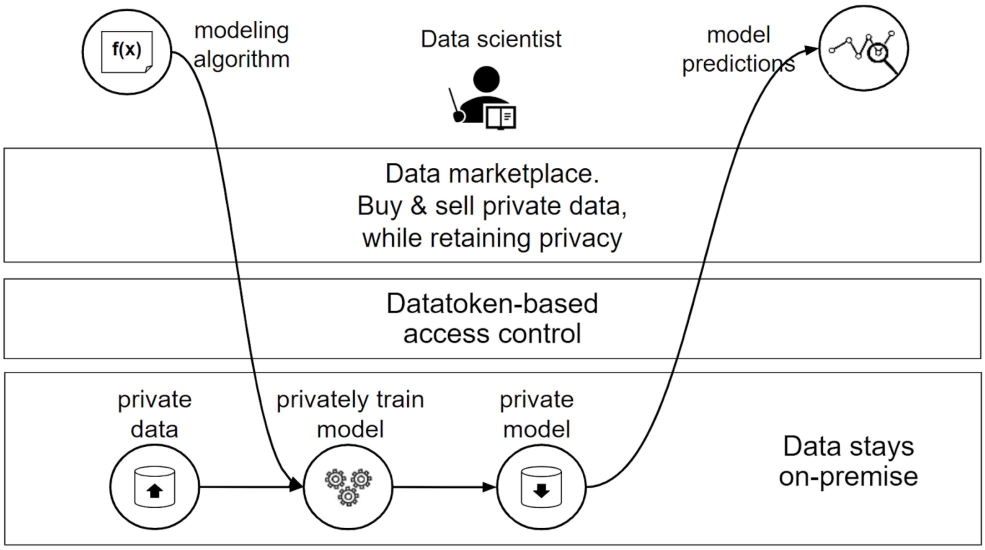

Figure 5: Ocean Compute-to-Data

3.7.3. Compute-to-Data Flow Variants

This section describes variants of how Compute-to-Data may be used. For further detail yet, we refer the reader to [McConaghy2020c].

In these variants, there is still model training next to the data:

- Alice is able to download the trained model, i.e. it can leave Bob's premises. This will happen if Bob believes the model has low risk of leaking personally identifiable information (PII), such as being a linear model or a small neural network.
- Alice learns a model across *many* data silos. This is "Federated Learning."

In the following, the compute that is run next to the data is *not* for training a model, but something else.

- A simpler "aggregating" function is run next to the data, such as an average, median, or simple 1-d density estimation. This means that Compute-to-Data is useful for simpler business intelligence (BI) use cases, in addition to more complex artificial intelligence (AI) use cases.
- An aggregating function is computed across *many* data silos. This is "Federated Analytics."
- A synthetic-data generation algorithm is run next to Bob's data. Alice downloads the synthetic data, which she then visualizes or trains models.
- A *hash* is computed for each {input variable, input value} combination of Bob's data, where the compute is done next to the data. Hashing naturally anonymizes the data. Alice downloads this hashed data and trains the model client-side. This is called "Decoupled Hashing."
- Random noise is added to Bob's dataset, sufficiently so for the data to be considered anonymized. Alice downloads this partly-randomized data and trains the model client-side. This is a variant of "Differential Privacy."

3.7.4. Share, or Monetize

Compute-to-Data is meant to be useful for data sharing in science or technology contexts. It's also meant to be useful for monetizing private data, while preserving privacy. This might look like a

paradox at first glance but it's not! The private data isn't directly sold; rather, specific access to it is sold, access "for compute eyes only" rather than human eyes. So Compute-to-Data in data marketplaces is an opportunity for companies to monetize their data assets.

3.7.5. Compute-to-Data Architecture

New actors. Ocean Protocol has these actors: **Data Providers**, who want to sell their data; **Data Consumers**, who want to buy data; and **Marketplaces**, to facilitate data exchange. Compute-to-Data adds a new actor, the **Compute Provider**. The Compute Provider sells compute on data, instead of data itself.

New Components. Ocean technology has several components. Operator Service and Operator Engine were introduced for Compute-to-Data.

- **Operator Service** — a microservice in charge of managing the workflow and executing requests. It directly communicates and takes orders from Provider (the data provider's proxy server) and performs computation on data, provided by Provider.
- **Operator Engine** — a backend service in charge of orchestrating the compute infrastructure using Kubernetes as a backend. Typically, the Operator Engine retrieves the workflows created by the Operator Service in Kubernetes. It also manages the infrastructure necessary to complete the execution of the compute workflows.

New Asset Type. Before, datasets were the only asset type in metadata (DDO). Compute-to-Data introduces a new asset type — algorithm, which is a script that can be executed on datasets.

For further detail, [Patel2020] provides a worked example and further references on Compute-to-Data.

3.7.6. Marketplaces and Compute-to-Data

Marketplaces can allow their users to publish datasets with Compute-to-Data enabled. Some marketplaces may even require it.

Marketplaces choose what exact compute resources they want to make available to their end users within a K8s cluster, even having them choose from a selection of different images and resources.

Likewise, marketplaces can choose and restrict the kind of algorithm they want to allow their users to run on the datasets in a marketplace.

3.7.7. Trusting Algorithms

Only trust in a narrow facet is required: does the algorithm have negligible leakage of PII? For example, a simple averaging function aggregates data sufficiently to avoid leaking PII. AI algorithms also aggregate information.

The data owner typically chooses which algorithms to trust. It's their judgement call. They might inspect the code and perhaps run it in a sandbox to see what other dependencies it causes, communications it invokes, and resources it uses. Therefore it's the same entity that risks private data getting exposed and chooses what algorithm to trust. It is their choice to make, based on their preference of risk vs. reward.

This also points to an opportunity for marketplaces of vetted algorithms: Ocean marketplaces themselves could be used, where liquidity provided is a proxy for quality and trust of the algorithm. Like with all pools, anyone can provide liquidity. This may be quite powerful, as it creates a "data science" side of the market.

3.7.8. Benefits of Compute-to-Data

Compute-to-Data has privacy benefits and other benefits as well:

- **Privacy.** Avoid data escapes, never leak personal or sensitive information.
- **Control.** Data owners retain control of their data, since the data never leaves the premises.
- **Huge datasets.** Data owners can share or sell data without having to move the data, which is ideal for very large datasets that are slow or expensive to move.
- **Compliance.** Having only one copy of the data and not moving it makes it easier to comply with data protection regulations like GDPR [GDPR2020].
- **Auditability.** Compute-to-Data gives proof that algorithms were properly executed, so that AI practitioners can be confident in the results.

3.7.9. Relation to other Privacy-Preserving Technologies

Compute-to-Data is complementary to other technologies such as encryption/decryption, Multi-Party Compute, Trusted Execution Environments, and Homomorphic Encryption. This compatibility and Ocean's "tools" framing helps make it easy to adopt. [McConaghy2020c] has details.

4. Subblock: OceanDAO

4.1. Introduction

The previous section described Ocean Tools to power a data ecosystem. We now discuss the next subblock in the Ocean System: OceanDAO. It addresses the questions:

- "How to kickstart growth?" and
- "How can the system not only self-sustain but actually improve over the decades, without guidance by a centralized actor?"

Our answer to this is OceanDAO: a DAO to implement community-proposed and curated grants. OceanDAO's income includes network rewards according to a distribution schedule. OceanDAO closes the loop with the rest of the Ocean System, such that growth in Ocean usage leads to more funding to teams, to work on further growing usage. This helps both short-term growth and long-term sustainability. It helps token holders, and would-be token holders *discover value* for the Ocean System (inspiration from [Warwick2020]).

OceanDAO was deployed live on Nov 30, 2020 [Napheys2020].

4.2. OceanDAO Components

OceanDAO takes income from Network Rewards and from Network Revenue.

This income is used to help drive improvement of core technology, apps, community development, and more.

OceanDAO has:

- An on-chain means for people to submit proposals.
- An on-chain means for the community to select proposals to fund, then fund them.
- An on-chain means to track progress and completion of each proposal.
- On-chain *funding* for proposals.
- Off-chain interfaces to the above.

4.3. Inspiration

OceanDAO is inspired by many Web3 sustainability projects, as surveyed in [McConaghy2020b]. DASH's grants program is of particular interest. It's been live since September 2015 [DashNexus2020]. Any allocated funds not spent are *burned*. This incentivizes a project to add not merely >0 value, but to add >*x* value where *x* is the rise in the value of each DASH token due to the reduction in total DASH tokens.

4.4. OceanDAO Funding Criteria

OceanDAO will use two criteria to judge projects for funding: sustainability/growth, and values. It's akin to a "double bottom line" of USA B-corps.

1. Help drive sustainability & growth of Ocean. This has the following sub-criteria.

- Project proposals must show how ROI (return on investment) could be far greater than 1.0 (if risky) or merely >1.0 (if less risky). This is so that ROI over all projects will average out to >1.0, the necessary condition for growth. ROI is in terms of potential impact on OCEAN, or proxies to that number such as Network Revenue or transaction count.
- Projects must have planning including milestones. This helps minimize risk. Teams that achieve milestones (including their spirit) will have a greater chance of receiving funding for future projects.
- Burn any funds not allocated in a given funding period.

2. **Project vision & execution must not go against Ocean Mission & Values**. This means that projects can either explicitly drive Ocean Mission & Values [McConaghy2018]; or be simply neutral (don't do anything against Mission & Values).

The filter for growth makes OceanDAO different than other grants DAOs, which tend to have more open-ended criteria. We believe this will be helpful in achieving ubiquity faster, and critical for long-term success because it ensures long-term funding as a function of usage.

We want to ensure long-term thinking in the voting; here are some inspirations using time-weighted voting power. In Conviction Voting [Emmett2019], vote increases with the amount of stake and the time staked for the vote. In Arweave "Profit Sharing Communities", votes are weighted by the number of tokens the voter holds, multiplied by the time they are willing to lock them [Arweave2020]. In Yearn.finance, weight is scaled by x/365, where x is days locked [Yearn2020].

We also aim to encourage project grantees to build up skin-in-the-game, via incentives to hold their tokens over longer periods. Inspiration comes from [Jain2020] [Fiskantes2020] [Coinlist2017] [Xdai2020b].

To help maximize the supply of data, we are considering a fraction of rewards for projects dedicated to unlocking data. An example project is the Ocean Data Farming program, which we describe below.

4.5. OceanDAO Implementation

Many tools for DAOs are emerging. At the smart contract level, these include Aragon [Aragon2020a], Moloch V2 [Turley2020], DAOStack [DaoStack2020], Colony [Colony2020], and Collab.Land [CollabLand2020]. Higher-level interfaces/apps for soft-signalling or voting include Snapshot, Boardroom, Pokemol, Daohaus, Discourse, Discord, and Telegram.

The DAO & governance space is evolving rapidly. Ocean will need to have flexibility to be able to learn over time. Therefore rather than deploying one "perfect" DAO a single time and hope that it stays perfect, we plan to "**bake slowly**" to give ample opportunity to change things.

- This means that Ocean Protocol Foundation (OPF) will deploy an initial OceanDAO, but for rapid learning in early days, change the deployed instance as often as every funding cycle. OPF may even deploy >1 DAO at once, in the spirit of DAOMesh [Lao2019].
- It also means that the funding itself will be manually disbursed at first.

Round 1 of OceanDAO was launched Nov 30, 2020 [Napheys2020]. Proposals are posted on a Ocean Port (a Discourse Forum), discussion has text chat on Ocean Discord and videochat in weekly Town Halls, and voting on Snapshot. The OceanDAO wiki is a unifying connector of these tools [OceanDAOwiki].

Over months or years, OceanDAO will accumulate learnings and stabilize more, at which point we will be able to crystallize the code and the funding to be truly decentralized, immutable, and censorship-resistant. This approach is in line with how Centrifuge [Stehlik2019] and others (review in [McConaghy2020b]) are approaching community DAOs. *Bake slowly.*

5. Subblock: Network Rewards

5.1. Network Rewards Schedule

5.1.1. Introduction

This section describes Ocean's network rewards schedule. Recall that network rewards feed into OceanDAO. First, *Bitcoin*'s network rewards schedule is:

$$F(H, t) = 1 - (0.5^{t/H})$$

where

- $F(H, t)$ is the fraction of all reward tokens that have been released after t years
- H is the half-life, in years. Half-life is the time taken for 50% of the remaining supply to be released.
- Bitcoin uses a half-life of 4 years.

The Bitcoin reward schedule has several nice features, including a fixed supply of tokens, more benefits to early movers, and a track record of being live for a decade.

For these reasons, Ocean uses Bitcoin's network rewards schedule, including half-life, as a starting point[13].

Our main additional challenge is: how to be resistant to instability in the early days of OceanDAO? E.g. if OceanDAO gets gamed or hacked. To address these, we "bake slowly" by **ratcheting up the amount of rewards over time, until stability is reached**. We start with manually-sourced grants directly from Ocean Protocol Foundation that aren't part of the 51%. With a bit of stability, we begin to draw on the 51% rewards. First, there's a 10% multiplier on the rewards curve, then 25% once we are more comfortable (6 months or less), then 50% after more comfortable yet, and finally 100%. The ability for manual intervention will be removed by the time the 100% multiplier is hit.

5.1.2. Ocean Network Reward Equations

This section describes the equations of network rewards, which implement the goals above. g(t) is the overall network rewards schedule. It's a piecewise model of four exponential curves.

[13] Using TokenSPICE modeling, we modeled both 4-year and 10-year half lives. The 4-year value had more appealing dynamics.

$$g(t) = \begin{cases} 0 & t < T0 \\ g_1(t) & T0 \leq t < T1 \\ g_2(t) & T1 \leq t < T2 \\ g_3(t) & T2 \leq t < T3 \\ g_4(t) & otherwise \end{cases}$$

Where $g_i(t)$ are pieces of the piecewise model chosen depending on *t*, and *Gi* are the values of $g_i(t)$ at the inflection points.

$$G1 = g_1(t = T1);\ G2 = g_2(t = T2);\ G3 = g_3(t = T3)$$

$$g_1(t) = M1 * f(t - T0)$$

$$g_2(t) = M2 * f(t - T0) - M2 * f(T1 - T0) + G1$$

$$g_3(t) = M3 * f(t - T0) - M3 * f(T2 - T0) + G2$$

$$g_4(t) = (1 - G3) * f(t - T3) + G3$$

Mi is a multiplier for a given interval *i*. *f(t)* is the value of *F(H,t)* assuming a constant *H*. *F(H,t)* is the base exponential curve. The units of *H* and *t* are years.

$$f(t) = F(H,t)$$

$$F(H,t) = 1 - (0.5^{\ t/H})$$

The pieces of the model are a function of t, which are parameterized with *Ti*. The units of *Ti* are years.

$$T0 =< time\ interval\ after\ network\ launch\ for\ network\ rewards >$$

$$T1 = T0 + 0.5$$

$$T2 = T1 + 0.5$$

$$T3 = T2 + 0.5$$

The following parameter values are chosen: *H*=4 years; *M1*=0.10; *M2*=0.25; *M3*=0.5.

5.1.3. Ocean Network Reward Curves

This section shows how network rewards are dispensed over time. It assumes *T0*=0.0 years and a 0.5 year interval to hitting *T1*, *T2*, and finally *T3*. These values may change in practice due to implementation. Figure 6 shows how network rewards will be dispensed over the next two years. One can readily see the ratchet every 6 months.

We had contemplated various designs for network reward schedules, and ran each candidate through agent-based simulation [TokenSPICE2020]. The best-performing design was chosen for the final design, which has been presented here.

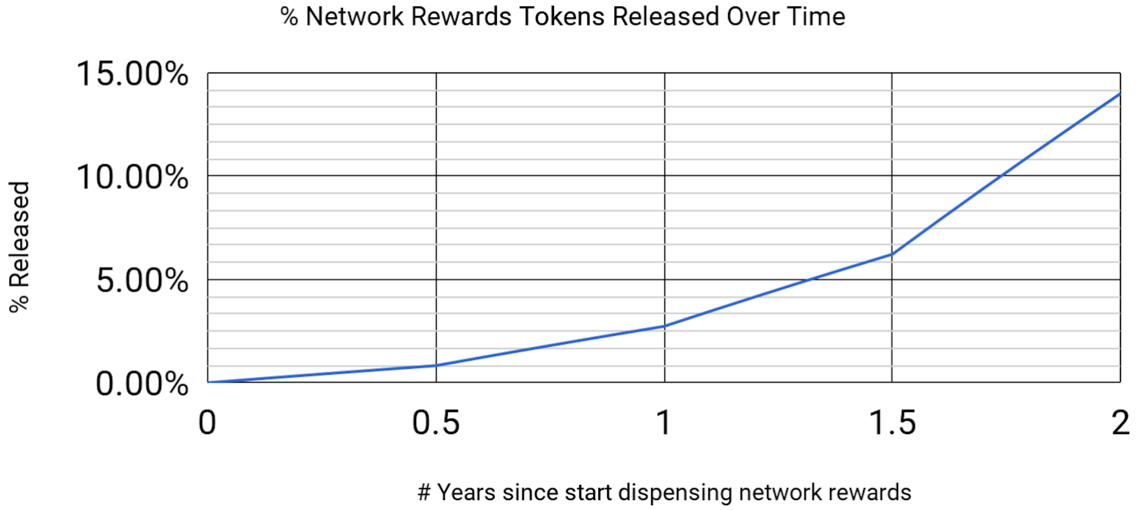

Figure 6: % network rewards tokens released over the next 2 years

Figure 7 shows network rewards over 25 years. The curve is shaped much like Bitcoin's, except the initial years of ramp-up.

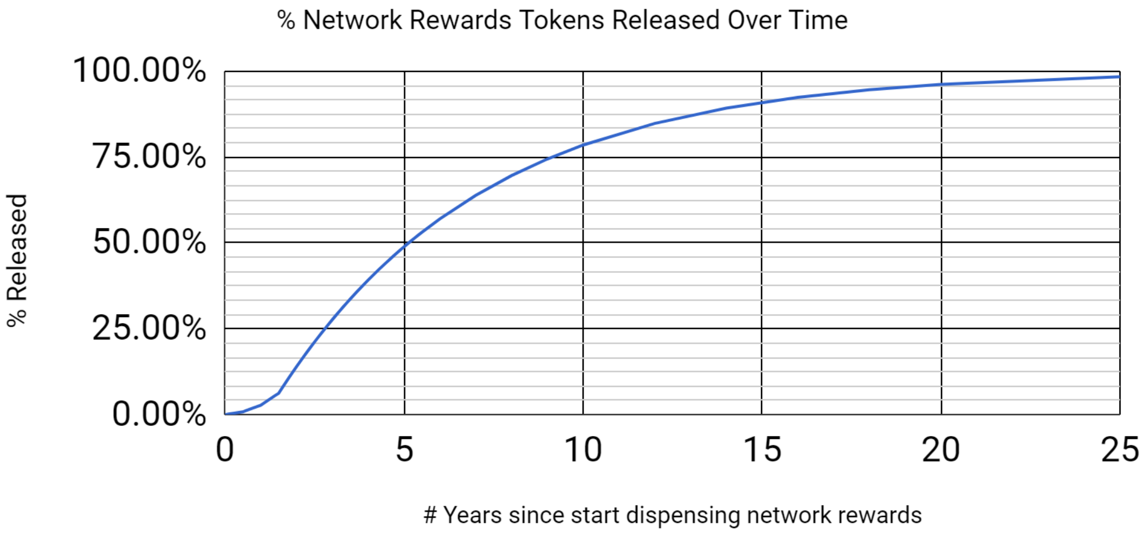

Figure 7: % network rewards tokens released over the next 25 years

Table 1 shows network rewards going to 50 years. It takes just over 5 years to get to 50% dispensed.

Table 1: % network rewards released over the next 50 years

	T0	T1	T2	T3								
# Years	**0**	**0.5**	**1**	**1.5**	**2**	**2.5**	**5**	**10**	**15**	**20**	**25**	**50**
% Re-leased	0%	0.83%	3%	6.2%	14.0%	21.1%	48.9%	78.5%	91.0%	96.20%	98.40%	99.98%
% Left	100%	99.17%	97%	93.8%	86.0%	78.9%	51.1%	21.5%	9.0%	3.80%	1.60%	0.02%

6. Ocean Data Farming

6.1. Overview

Ocean **Data Farming** is a program to incentivize a supply of **relevant and high-quality data assets.**

ODF aims to maximize the **data supply reward function (RF)**, which is a function of liquidity added, dataset usage, and potentially more. In other words, liquidity providers (LPs) can earn extra OCEAN rewards for providing liquidity, and amount earned will be multiplied by usage of the datasets they stake on, and more.

We will start with an initial RF and a program budget of OCEAN over a time interval (e.g. 3 months). Each week at a pre-set time, the program managers will:

- Off-chain, calculate contribution of each actor to the RF for that week (actor = Ethereum address).
- Manually airdrop OCEAN to each actor proportional to their contribution, the total being that week's budgeted OCEAN.
- Declare the RF for use in the following week, to fine-tune towards the most relevant and high-quality datasets, and away from "gaming" that is against the spirit of the program.

During the week, the program managers will:

- Gather feedback from the community on possible ways to improve the RF, using soft signalling[14] in Discord or similar. Later generations may use OCEAN token voting.

The OCEAN will budget may draw from OPF treasury and (if the community votes for it) OceanDAO funds.

This design allows the program managers to start simple, then to quickly learn and improve the RF every week. This design draws inspiration from the successful Balancer Liquidity Mining Program [Martinelli2020]. If the initial ODF run (over the time interval) is successful, it may be repeated.

6.2. Reward Function

This section describes a tentative initial RF. It may change before the program initiates.

RF_{ij} is the (non-normalized) contribution for actor *i* on datatoken *j*:

$$RF_{ij} = log_{10}(S_{ij} + 1) * log_{10}(D_j + 1)$$

where:

- S_{ij} = actor *i*'s OCEAN stake in data asset *j* = (actor's # BPTs[15] in datatoken *j*'s pool / total # BPTs in pool) * (# OCEAN in pool)[16]
- D_j = # times data asset *j* has been consumed in the last week (= # consume transfers to the datatoken *j*'s Provider)

[14] E.g. giving a thumbs-up or thumbs-down to questions posted in Discord.

[15] BPT = Balancer Pool Token. People receive BPTs proportional to the OCEAN or datatoken liquidity they add to a pool. Conversely, when they remove liquidity, they are essentially selling BPTs for OCEAN and datatokens.

[16] In the first iteration, rewards will only go to liquidity in Ocean-tweaked Balancer pools. Over time, we anticipate this to expand to other exchanges.

The first term is $log_{10}(S_{ij})$. It reflects the actor's belief in the relevance, quality, or potential usage of the datatoken. This incentivizes Providers to publish relevant data assets. We use $\log_{10}$ to level the playing field with respect to OCEAN whales; and so that OCEAN whales are incentivized to make a greater number of data assets available. This has theoretical roots in Kelly Betting: applying the log is the optimal strategy for an individual to maximize their utility in betting [KellyCriterion2017] [Simmons2017].

The second term, $log_{10}(D_j)$, is a direct measure for the usage of the data asset, and a proxy for its relevance or quality. We use $\log_{10}$ to incentivize actors to focus on usage of a greater number of assets. The +1 is to ensure there's a reward even if the dataset is used once in a given week.

This RF_{ij} can be summarized as a binding of *predicted* popularity * *actual* popularity in terms of their orders-of-magnitude. As discussed, this is the *initial* RF. We anticipate feedback from the community to further improve it.

To help long-term thinking and drive value to the protocol, we are considering lockup periods to ensure that rewards are not immediately sold by traders [Jain2020]. It could even be "longer lockup, longer reward" [Fiskantes2020], like in Filecoin's ICO [Coinlist2017] or xDai Chain's staking incentives [Xdai2020b].

Astute readers may recognize this design from the 2018 and 2019 versions of the Ocean whitepaper, to maximize the supply of data. That's correct. There are two main differences. We've changed it from smart contracts in the core of the system, into a program to catalyze growth run manually and off-chain for rapid learning and iterations. Also, the rewards formula has a simpler implementation, due to using staking inside datatoken pools and datatoken transfer calls.

7. On OCEAN Token Model

7.1. Introduction

This section summarizes the OCEAN token from a crypto-economic perspective.

7.2. Token Value ∝ Volume

To ensure long-term funding for the Ocean community, **we want OCEAN health to increase as usage volume increases. Ocean does this; here's how.**

- Here's the loop: more usage of Ocean tools in the Data Ecosystem leads to more OCEAN being staked, leading to more OCEAN demand, growing $OCEAN.
- More usage also leads to more Network Revenue, which goes to (i) burning and (ii) OceanDAO. Burning OCEAN reduces supply, to grow $OCEAN. Funds go through OceanDAO to workers who have the mandate to grow usage of Ocean tools. And the loop repeats.

7.3. Work Token (and Governance Token)

"Work" is when actors perform services for the token ecosystem that add value to the system[17]. Ocean incentivizes for work to happen in OceanDAO governance, Ocean Workers block, and Ocean marketplace tools. This gives OCEAN some **work token** dynamics [Samani2018].

OceanDAO Governance. In Ocean, governance is about directing the flow of resources towards the teams and projects that offer the best chance for growth. OCEAN holders are incentivized to do work

[17] This is more general than the idea that one must buy a work token to have access to revenue streams in the network (like a NYC taxi medallion).

to learn more about each team and project proposal, and then use OceanDAO to vote for the most promising teams/projects (subject to Ocean mission & values). If there aren't enough value-adding projects for the OCEAN funds available, remaining OCEAN gets burned; this therefore sets a baseline for the value-add needed. Note that governance is *not* about specific decisions for protocols or software, it's at a higher level and more directly about value creation.

Ocean Workers. Paid by OceanDAO, these teams do a variety of work, including: improving core software (datatokens, etc), building and improving applications, outreach and community building, and unlocking data for use in Ocean. All of these activities drive value to the Ocean ecosystem (on average) because the work has been filtered by OceanDAO towards the value-adding activities.

Ocean marketplace tools. There are several ways for the Ocean ecosystem to gain value via "work" inside the "data ecosystem" box of Figure 1.

- **More liquidity** in datatoken pools gives lower slippage, for a better data buyer experience, which in turn drives volume, and in turn Ocean value. The promise of **LP transaction fees** incentivizes people to add OCEAN liquidity to OCEAN-datatoken pools.
- **Curation of assets** (=amount staked in the corresponding datatoken pool) allows data buyers to surface the best assets more easily, which in turn drives volume, and in turn Ocean value. To have the best returns, LPs are incentivized to **seek datatoken pools with the highest volumes**, which tend to be the ones with the best datasets. That is, they're curating.
- **Referrals** bring new users to Ocean tools, which in turn drives volume, and in turn Ocean value. Because LPs / stakers / curators get fees on transactions, they are incentivized to refer people to the pools that they have staked in.
- **More marketplaces** with good volumes drives total Ocean volume, and in turn Ocean value. The promise of **marketplace fees** incentivizes people to run their own marketplaces and grow their volumes.

Ocean Data Farming. This program incentivizes workers to add data assets, use data assets, and add liquidity to data assets. Future versions may also use OCEAN-based voting to govern updates to their reward formulae.

7.4. Burning

First, 5% of all Network Revenue is burned. This ensures that OCEAN holders will always see some deflationary actions, and as Network Revenue increases, deflation increases. The 5% value was chosen based on running various possible values in agent-based simulation [TokenSPICE2020].

Second, for a given funding period, OceanDAO burns all non-allocated funds.

7.5. User Acquisition

Special OceanDAO projects like Data Supply Mining are not core to the Ocean token design itself, but spend some OCEAN resources in order to help acquisition of users, data, liquidity, and the like.

7.6. On OCEAN as Unit-of-Exchange

OCEAN can be used as a unit-of-exchange to buy and sell datatokens, but it's not mandated. Being ERC20 tokens, datatokens may be transferred in return for whatever payment is desired, such as fiat, DAI, or ETH. Community participants may add capabilities like this as they see fit.

OCEAN is the default unit-of-exchange in Ocean Market and in Ocean libraries, for both fixed-price and auto-pricing scenarios (the latter also defaults to staking OCEAN). While Ocean Market transacts in OCEAN, it also displays the price in USD and EUR. Community participants may build data marketplaces that transact in fiat or other crypto-tokens.

The essay [McConaghy2020g] elaborates on the Ocean token model further.

8. Ocean Applications

8.1. Decentralized Orchestration

Here's an example compute pipeline: input raw training data → clean the data → store cleaned data → build model → store model → input raw test data → run predictions → store predicted result y_test. Leveraging Ocean, a developer can write Solidity code [Solidity2020] to define a pipeline and execute it, i.e. do "decentralized orchestration":

1. Write a smart contract that uses various data services. There's a datatoken for access control of each data service.
2. Deploy the smart contract to Ethereum mainnet.
3. For each different datatoken, do a tx that approves the required amount of datatokens to the SEA smart contract.
4. Finally, do a transaction that makes the actual call to the smart contract.

With a Set Protocol, ERC998 or similar to have a "basket" datatoken that holds all the necessary sub-datatokens in the compute pipeline, step 3 becomes simpler yet.

These Solidity scripts can be seen as "Service Execution Agreements" (SEAs) [DeJonghe2018], a riff on "Service Level Agreements" for centralized orchestration.

A major sub-application is decentralized federated learning (decentralized FL). In traditional FL, a centralized entity (e.g. Google) must perform the orchestration of compute jobs across silos. So, PII can leak to this entity. OpenMined [OpenMined2020] could decentralize orchestration, using Ocean to manage computation at each silo in a more secure fashion [McConaghy2020c].

8.2. Data Wallets: Data Custody & Data Management

Data custody is the act of holding access to the data, which in Ocean is simply holding datatokens in wallets. Data management also includes sharing access to data, which in Ocean is simply transferring datatokens to others.

With datatokens as ERC20 tokens, we can leverage existing ERC20 wallets. This includes browser wallets (e.g. Metamask), mobile wallets (e.g. Argent, Pillar), hardware wallets (e.g. Trezor, Ledger), multi-sig wallets (e.g. Gnosis Safe), institution-grade wallets (e.g. Riddle & Code), custodial wallets (e.g. Coinbase Custody), and more.

ERC20 wallets may get tuned specifically for datatokens as well, e.g. to visualize datasets, or long-tail token management (e.g. holding 10,000 different datatoken assets).

Existing software could be extended to include data wallets. For example, Brave browser has a built-in crypto wallet that could hold datatokens. There could be browser forks focused on datatokens, with direct connection to user browsing data. Integrated Development Environments (IDEs) for AI like Azure ML Studio [Azure2020] could have built-in wallets to hold & transfer datatokens for training data, models as data, and more. Non-graphical AI tools could integrate; such as scikit-learn or TensorFlow Python libraries using a Web3 wallet (mediated with Ocean's Python library).

As token custody continues to improve, data custody inherits these improvements.

8.3. Data Auditability

Data auditability and provenance is another goal in data management. Thanks to datatokens, blockchain explorers like Etherscan [Etherscan2020] now become data audit trail explorers.

Just as CoinGecko [CoinGecko2020] or CoinMarketCap [CoinMarketCap2020] provide services to discover new tokens and track key data like price or exchanges, we anticipate similar services to emerge for datatokens. CoinGecko and CoinMarketCap may even do this themselves, just as they've done for DeFi tokens.

8.4. Data DAOs: Data Co-ops and More

Decentralized Autonomous Organizations (DAOs) [DAO2020] help people coordinate to manage resources. They can be seen as multi-sig wallets, but with significantly more people, and with more flexibility. DAO technology is maturing well, as we reference in the OceanDAO section. A "data DAO" would own or manage datatokens on behalf of its members. The DAO could have governance processes on what datatokens to create, acquire, hold, sell / license, and so on.

Here are some applications of data DAOs:

Co-ops and Unions (Collective Bargaining). Starting in the early 1900s, thousands of farmers in rural Canada grouped into the SWP [SWP2020] for clout in negotiating grain prices, marketing grain, and distributing it. Labour unions have done the same for factory workers, teachers, and many other professions. In [Ibarra2018], the authors suggest that data creators are currently getting a raw deal, and the solution is to make a labour union for data. A data DAO could be set up for collective bargaining, as a "data coop" or "data union". For example, there could be a data coop with thousands of members for location data, using FOAM proof-of-location service [McConaghy2020a].

Manage a single data asset. There could be a DAO attached to a single data asset. One way is: create a Telegram channel dedicated to that dataset. You can only enter the Telegram channel if you have 1.0 of the corresponding datatokens (inspired by Karma DAO [Lee2020]). This can also be for Discord, Slack, or otherwise.

Datatoken pool management. There could be a data DAO to manage a datatoken pool's weights, transaction fees, and more, leveraging Balancer Configurable Rights Pools [Balancer2020b] (inspired by PieDAO which does this for a pool of DeFi assets [Delmonti2020]).

Index Funds for Data Investments. Using e.g. Melon [Melon2020], an investment product can be constructed to allow people to buy a basket of data assets with the current plethora of mutual and index funds as a guide.

8.5. Permissioned Group-Restricted Access in Data Exchanges

In this operational model, "membership rules" apply for a group. These rules are governed by a verifiable credential, a TCR, a custom whitelist, a DAO, or otherwise. These membership rules enable the following applications for data sharing.

Contests, Hackathons, Impromptu Collaborations. A group of hackers or data scientists self-organize to try to solve an AI problem, such as a Kaggle competition or a hackathon. They want to be able to easily access each others' data and compute services as they progress, especially if they are working remotely from each other.

Regulatory Sandboxes. A government wants to give a means for organizations to run in a "monitored" regulatory environment that the government can observe. The organizations are vetted by the government and may have access to specially designated government compute services.

Enterprise data access. An enterprise might make some of its data available to only its employees, but want to be able to use Ocean services available in the broader network.

Sharing autonomous driving data. Individuals in each membership company of MOBI [MOBI2020] need to access automotive data from any one of the MOBI member companies. It could be time-consuming and error-prone to specify permission for each member company individually. Furthermore, those permissions will fall out of date if MOBI members are added or removed; and updating the permissions one organization at a time could also be time-consuming or error-prone. This involves two levels of permissions: access of member companies into MOBI, and access of individuals in each member company (enterprise).

Sharing medical data. Researchers on European soil that wish to directly access German medical data need to demonstrate that they have been accredited by appropriate authorities. This will usually be through their hospital or university. There could be thousands of researchers accessing the data. As with automotive data, it will be time-consuming and error prone to specify and update permissions for each of these thousands of researchers. This may be two levels of permissions (hospital/university into EU authority; individual into hospital/university), or it may be among hospitals and universities in a more networked fashion.

Sharing financial data (while preserving privacy). Small and medium-sized credit unions in the U.S. have a challenge: they don't have large enough datasets to justify using AI. Since the credit unions don't compete with each other, they would find great value to build AI models across their collective datasets.

8.6. Unlocking Latent Data of Individuals & Enterprises

Specialized apps could get built to tokenize and earn from latent data assets of individuals or organizations.

Individuals. For example, an app that you install on your phone, you give permissions to access data (Compute-to-Data for privacy), then it auto-creates an AMM market for your data. After that you earn money from the data on your phone. In a mashup with personal tokens [Chaturvedi2020], these would be your sovereign "personal datatokens" and you could launch your data as an "Personal Data Offering" (PDO).

One could even launch multiple PDOs. There would be one datatoken for each data type generated - smartphone data, smartwatch data, browser data, shopping data, and so on. Then, bundle the datatokens into a collection of personal datatokens (composable datatokens). One could even have a personal data marketplace[18].

One could do the same with a browser plug-in, to sell the latent data on your browser (cookies, bookmarks, browsing history). New service firms could emerge to help enterprises tokenize and earn from their massive internal data troves.

These data assets might be sold one person or entity at a time. Or, they could be brought into a DAO to pool resources for more marketing and distribution muscle.

Enterprises. Large enterprises have massive datasets. They know their data is valuable: they spend millions annually to help protect it and insure it due to hacks. But what if rather than data being a *liability*, data was *assets* on enterprises' books? Ocean offers this possibility, by making it easy to turn the internal data into fungible assets (via datatokens), automatically discover the prices (via AMMs),

[18] Thanks to Simon Mezgec for this suggestion.

while preserving privacy and control (via Compute-to-Data). Data not only becomes a new line of revenue, it becomes a *financial asset* that can be borrowed against to fund growth, and more.

8.7. Data Marketplaces

Earlier, we described Ocean Market, an out-of-the-box data marketplace that Ocean offers. Here are some ways that forks of Ocean Market could differentiate.

- **Focus on a given vertical.** For example, the Ocean-based dexFreight data marketplace focuses on the logistics vertical. Other verticals include health, mobility, and DeFi.
- **Focus on private data.** A marketplace focusing on Ocean compute-to-data, with features for curation of compute algorithms.
- **Different fee structure.** When a purchase happens in Ocean Market, the default 0.1% fee goes fully to LPs. Variants include: marketplace operator gets a cut, referrers get a cut, charge higher %, charge a flat fee.
- **Novel pricing mechanism.** Many price discovery mechanisms are possible [Kuhn2019a] [Kuhn2019b], including royalties (a % of sales), English / Dutch / Channel auctions [Azevedo2018], or income share agreements like Bowie or Dinwiddie bonds [McConaghy2019d]. This may include a novel initial distribution mechanism, as the Initial Data Offerings section elaborates.
- **Different payment means.** Ocean Market takes OCEAN. Variants could take in fiat, DAI, ETH, etc. (Ocean Market may support these directly over time as well. PRs are welcome!)
- **Decentralized dispute resolution** using Aragon Court [Aragon2020b] or Kleros [Kleros2020].
- Different Balancer weighting schemes. For example, make the 90/10 weights shift to 50/50 over time (Liquidity Bootstrapping). Or, surge pricing pools, which have higher fees when there is more demand for liquidity. These could be implemented with Configurable Rights Pools. PRs for Ocean Market are welcome!
- Different DEXes. Ocean Market currently uses a tweak of Balancer AMMs for lower gas costs. 3rd party marketplaces may use the original Balancer deployment, Uniswap, Bancor, Kyber, or other.

Beyond forks of Ocean Market, there is an larger variety of possible marketplaces. Here are some Web2 and Web3 variants.

- **AMM DEXes.** This could be a Uniswap or Balancer-like webapp to swap datatokens for DAI, ETH, or OCEAN. It could also have something like pools.balancer.exchange to browse across many datatoken pools.
- **Order-book DEXes**. It could use 0x, Binance DEX, Kyber, etc. It could leverage platform-specific features such as 0x's shared liquidity across marketplaces.
- **Order-book CEXes.** Centralized exchanges like Binance or Coinbase could readily create their own datatoken-based marketplaces, and to kickstart usage could sell datasets that they've generated internally.
- **Marketplaces in AI tools.** This could be an AI-oriented data marketplace app embedded directly in an AI platform or webapp like Azure ML Studio or Anaconda Cloud. It could also be an AI-oriented data marketplace as a Python library call, for usage in any AI flow (since most AI flows are in Python). In fact, this is already live in Ocean's Python library.
- **"Nocode" Data Marketplace builder.** Think Shopify [Shopify2020] for data marketplaces, where people can deploy their own data marketplaces in just a few clicks.

8.8. Initial Data Offerings (IDOs)

In an IDO, people or organizations can launch data assets using the technology tools & marketing techniques to launch other tokens (e.g. for Initial Coin Offerings and Initial Exchange Offerings).

2017 brought a craze of Initial Coin Offerings (ICOs), for better and for worse. Great efforts were put into designing mechanics of token distributions, and marketing and legals. A lot was learned from that era, and the learning has continued.

Here are some innovations since then. Vitalik Buterin suggested **DAICOs**, which leverages DAO technology to decentralize fundraising effort [Buterin2018]. Fabian Vogelsteller suggested **reversible ICOs** (rICOs), which gave investors the ability to pull their funds out, to minimize their risk [Vogelsteller2020].

Binance and other CEXes offer **Initial Exchange Offerings** (IEOs) which hold a fixed-price token sale, followed immediately by trading on the exchange. UMA and others have conducted **Initial DEX Offerings** using an AMM as the first market for their token [Uma2020]. Balancer's **Liquidity Bootstrapping Pools** (LBPs) [McDonald2020] refine this by slowly releasing more of the token over several months while simultaneously increasing its weight in the pool (towards 50-50 liquidity).

Unisocks [Unisocks2020] and Karma DAO [Lee2020] each have a combination of **bonding curve + AMM**. The bonding curve [Rouviere2017] acts as a primary market, where price increases with each token minted. The AMM acts as a secondary market. YFI (and Bitcoin!) had no **"offering" at all**. Rather, tokens got distributed to users for doing "work" to add value to the system. The philosophy is "earned, not printed" [@Bitcoin2020]. Incidentally, this is how the majority of OCEAN are distributed too, curated by OceanDAO.

An Initial Data Offering (IDO) can use any of these techniques. A pragmatic starting point is **Initial DEX Offering** via an AMM. This is simple, and well-suited to long-tail tokens like datatokens. Ideally, there's software to make launching such datatokens easy; call it an **IDO Launchpad**. Ocean Market makes it easy to publish a datatoken and create a Balancer AMM all at once; therefore **Ocean Market is the first IDO Launchpad**. We envision other IDO variants in the future, from the list above and new innovations. We hope to see more IDO Launchpads created by ambitious teams.

8.9. Data as an Asset Class for DeFi

The data economy is already 377B€ for Europe alone [EuroComm2019]. Tokenized data assets have great promise to grow the size of DeFi assets under management.

Data can be securitized and used as collateral [McConaghy2019d]. An example is Bowie Bonds, where a fraction of David Bowie's IP (intellectual property) licensing revenue was paid to bondholders. Data is IP. To use it as a financial asset, one must price it. In Bowie's case, the value was established from previous years' licensing revenue. Alternatively, we can establish price by selling data assets in data marketplaces.

As such, data is an asset class. With datatokens, we can onboard more more data assets into each major DeFi service types:

- Data assets can be used as collateral in **stablecoins** and **loans**, therefore growing total collateral.
- Data assets bought and sold in **DEXes** and **CEXes** contributes to their $ volume and assets under management (AUM).
- There can be insurance on data assets. As described above, there can be data DAOs, data baskets, and more.

8.10. Data to Optimize DeFi Returns

We can close the loop with data helping DeFi, and vice versa. Specifically: **data can improve decision-making in DeFi to optimize returns.** This will catalyze the growth of DeFi further. Here are some examples:

- **Yield farming.** Data can improve the automated strategies to maximize APR in yield farming. Think yearn.finance robots, but optimized further.
- **Insurance.** Data to lower the risk models in insurance.
- **Loans.** Better prediction of default for under-collateralized loans.
- **Arb bots.** More data for higher-return arbitration bots.
- **Stablecoins.** Assessment of assets for inclusion in stablecoins.

Data-powered loops. DeFi looping techniques further boost returns. For each of the examples above, we envision *loops* of buying more data, to get better returns, to buy more data, and so on. To go even further, we could apply this to data *assets* themselves.

8.11. Data Management Platforms for Smart Cities and More

In 2001, the government of Estonia rolled out a data management platform called X-Road [XRoad2020]. It then deployed an identity system on top; each citizen received an identity card with a digital signature. Since then, Estonia has rolled out apps for elections, health, taxes, parking, lawmaking, E-Residency and two dozen more government apps, plus 3rd-party apps like banking [eEstonia2020].

Both Ocean and X-Road can be used as digital infrastructure for smart cities' data sharing. X-Road can be seen as a smart city example, since the majority of Estonians live in Tallinn. X-Road has a longer history, but has more centralized control and requires dev-ops effort. While Ocean is younger, by using a global permissionless infrastructure, it has lighter dev-ops requirements. We envision a future with both X-Road and Ocean-based data sharing in smart cities.

Ocean framed as a data management platform can be used not only for data sharing within a city (across citizens), but also within a province/state (across cities), within a nation (across provinces), within an international initiative (across nations, e.g. GAIA-X [GAIAX2020]), within a company (across employees), and within a multinational enterprise (across national offices).

8.12. Composable Datatokens

Datatokens can be composed into bundles, sets, or groups using ERC998 [Lockyer2018], Set Protocol [Set2020], Melon Protocol [Melon2020], or others. This helps for the following use cases:

- **Group across time.** Package each 10-min chunk of data from the last 24 hours into a single token.
- **Group across data sources.** Package 100 data streams from 100 unique Internet-of-Things (IoT) devices, as a single token.
- **Data baskets for asset management.** Group together 1000 datasets that each have individual (but small) value, to sell as a single asset to others wanting to hold data assets.
- **Data indexes.** Track the top 100 data assets and make it easy for others to invest in those as a single asset, similar to today's index funds.
- **On-chain annotations to metadata.** Use ERC998 in a bottom-up setting to "attach" tags or other information to the data asset. Uses include: reputation given by a marketplace's users,

quality as computed by a marketplace's algorithms, input training data vs output vs a model, industry verticals, and more.

The essay [McConaghy2019d] elaborates further.

9. Conclusion

This paper presented Ocean Protocol. Ocean tools help developers build marketplaces and other apps to privately & securely publish, exchange, and consume data. The tools offer an on-ramp and off-ramp for data assets into crypto ecosystems, using datatokens. Composability gives many application opportunities, including data wallets, data marketplaces, data DAOs, and more. Ocean Market is a live reference community marketplace that natively integrates Balancer AMMs, to facilitate "Initial Data Offerings".

Ocean tools are encapsulated in a broader system designed for long-term growth of an open, permissionless Web3 Data Economy. Ocean Data Farming incentivizes a supply of quality data. A key piece is OceanDAO to fund software development, outreach. OceanDAO is funded by revenue from Ocean apps and from Ocean's network rewards.

10. References

[1inch2020] 1Inch Team, "Balancer Pool with STA Deflationary Token Incident", 1inch Blog, June 29, 2020, https://medium.com/@1inch.exchange/balancer-hack-2020-a8f7131c980e

[3Box2020] 3Box Team, "3Box: User Data Cloud", 3Box Homepage, last accessed Aug 19, 2020, https://3box.io/

[Aragon2020a] "Aragon", homepage, last accessed Aug. 20, 2020, https://aragon.org/

[Aragon2020b] "Aragon Court", homepage, last accessed Aug. 20, 2020, https://aragon.org/court

[Arweave2020] "Profit Sharing Communities", Arweave homepage, last accessed Aug. 29, 2020, https://staging.arweave.org/profit-sharing-communities

[Azevedo2018] Eduardo M. Azevedo, David M. Pennock, and E. Glen Weyl, "Channel Auctions", Aug 31, 2018, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3241744

[Azure2020] Azure ML Team, "Microsoft Azure Machine Learning Studio", homepage, last accessed Aug. 19, 2020, https://studio.azureml.net/

[Balancer2019] Balancer Team, "Interest-Bearing Stablecoin Pools Without Impermanent Loss", Balancer blog, Oct. 30, 2019, https://balancer.finance/2019/10/30/interest-bearing-stablecoin-pools-without-impermanent-loss/

[Balancer2020a] "Balancer", homepage, last accessed Aug. 21, 2020, https://balancer.finance

[Balancer2020b] Balancer Team, "Configurable Rights Pool", GitHub Repository, published Aug, 2020, https://github.com/balancer-labs/configurable-rights-pool

[Banko2001] M. Banko and E. Brill, "Scaling to Very Very Large Corpora for Natural Language Disambiguation", Proc. Annual Meeting on Association for Computational Linguistics, July, 2001, http://www.aclweb.org/anthology/P01-1005

[@Bitcoin2020] @Bitcoin, Twitter, Apr. 4, 2020 https://twitter.com/Bitcoin/status/1246482664376414209

[Buterin2013] Vitalik Buterin, "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform", Dec. 2013, https://ethereum.org/en/whitepaper/

[Buterin2018] Vitalik Buterin, "Explanation of DAICOs", Eth.Research Blog, Jan. 2018, https://ethresear.ch/t/explanation-of-daicos/465

[Buterin2020] Vitalik Buterin, Twitter, Feb. 18, 2020, https://twitter.com/VitalikButerin/status/1229819782230241280

[Chainlink2020] "Chainlink", Homepage, last accessed Aug. 20, 2020, https://chain.link/

[Chaturvedi2020], Vinamrata Chaturvedi, "The Man Who Tokenized Himself Gives Holders Power Over His Life", June 30, 2020, https://www.coindesk.com/man-who-sells-himself-now-wants-buyers-to-control-his-life

[CollabLand2020] "Collab.Land", Homepage, last accessed Aug. 20, 2020, http://collab.land/

[CoinGecko2020], CoinGecko Team, CoinGecko website, last accessed Aug. 19, 2020 https://www.coingecko.com

[CoinMarketCap2020], CoinMarketCap Team, CoinMarketCap website, last accessed Aug. 19, 2020, https://www.coinmarketcap.com

[Coinlist2017], Coinlist Team, "Filecoin Token Sale Economics", fall 2017, https://coinlist.co/assets/index/filecoin_2017_index/Filecoin-Sale-Economics-e3f703f8cd5f644aecd7ae3860ce932064ce014dd60de115d67ff1e9047ffa8e.pdf

[Colony2020] "Colony", Homepage, last accessed Aug. 20, 2020, https://colony.io/

[CosmosSdk2020] "Cosmos SDK Documentation", homepage, last accessed Aug. 19, 2020, https://docs.cosmos.network/

[CRUD2020] "Create, read, update, and delete", Wikipedia, last accessed Aug. 19, 2020, https://en.wikipedia.org/wiki/Create,_read,_update_and_delete

[DAO2020] "Decentralized Autonomous Organization", Wikipedia, last accessed Aug. 19, 2020, https://en.wikipedia.org/wiki/Decentralized_autonomous_organization

[DaoStack2020] "DaoStack", homepage, last accessed Aug. 20, 2020, https://daostack.io/

[DashNexus2020] "DashNexus", homepage, last accessed Aug. 20, 2020, https://app.dashnexus.org

[DeJonghe2018] Dimitri de Jonghe, "Exploring the SEA: Service Execution Agreements", Ocean Protocol Blog, Nov. 30, 2018, https://blog.oceanprotocol.com/exploring-the-sea-service-execution-agreements-65f7523d85e2

[Delmonti2020] Alessio Delmonti, "Introducing PieDAO, the asset allocation DAO", Mar. 3, 2020, https://medium.com/piedao/introducing-piedao-the-asset-allocation-dao-1af9eec5ee4

[DID2019] "Decentralized Identifiers(DIDs) v0.11: Data Model and Syntaxes for Decentralized Identifiers", W3C Community Group, Draft Community Group Report 06 February 2019, last accessed Feb. 28, 2019, https://w3c-ccg.github.io/did-spec/

[Economist2017] "The World's Most Valuable Resource is No Longer Oil, But Data", *The Economist*, May 6, 2017, https://www.economist.com/news/leaders/21721656-data-economy-demands-new-approach-antitrust-rules-worlds-most-valuable-resource

[eEstonia2020] "E-Estonia", homepage, last accessed Aug. 19, 2020, https://e-estonia.com/

[Emmett2019] Jeff Emmett, "Conviction Voting: A Novel Continuous Decision Making Alternative to Governance", July 3, 2019, https://medium.com/giveth/conviction-voting-a-novel-continuous-decision-making-alternative-to-governance-aa746cfb9475

[Etherscan2020] "Etherscan", homepage, last accessed Aug. 21, 2020, https://etherscan.io

[EthOptimism2020] Ethereum Optimism Team, "Optimistic Virtual Machine Alpha", Ethereum Optimism Blog, Feb. 12, 2020, https://medium.com/ethereum-optimism/optimistic-virtual-machine-alpha-cdf51f5d49e

[EuroComm2019] European Commission, "Building a Data Economy - Brochure", Sep. 2019, https://ec.europa.eu/digital-single-market/en/news/building-data-economy-brochure

[ERC721] "ERC-721", Homepage, last accessed Aug. 19, 2020, http://erc721.org/

[ERC1167] Peter Murray, Nate Welch, and Joe Messerman, "EIP-1167: Minimal Proxy Contract", Ethereum Improvement Proposals, no. 1167, June 2018, https://eips.ethereum.org/EIPS/eip-1167

[Fiskantes2020] @Fiskantes, Twitter, Aug 12, 2020, https://twitter.com/fiskantes/status/1293372786002538496?s=11

[GAIAX2020] "GAIA-X", homepage, last accessed Aug. 21, 2020, https://www.data-infrastructure.eu/

[GDPR2020] "General Data Protection Regulations", Wikipedia, last accessed Aug. 21, 2020, https://en.wikipedia.org/wiki/General_Data_Protection_Regulation

[Goldin2017] Mike Goldin, "Mike's Cryptosystems Manifesto", Google Docs, 2017, https://docs.google.com/document/d/1TcceAsBlAoFLWSQWYyhjmTsZCp0XqRhNdGMb6JbASxc/edit

[Graham2005] Paul Graham, "How to Start a Startup", Paul Graham Blog, March 2005, http://www.paulgraham.com/start.html?viewfullsite=1

[Halevy2009] Alon Halevy, Peter Norvig, and Fernando Pereira, "The Unreasonable Effectiveness of Data", IEEE Intelligent Systems 24(2), March-April 2009, https://research.google.com/pubs/archive/35179.pdf

[Ibarra2018] Imanol Arrieta Ibarra et al, "Should We Treat Data as Labor? Moving Beyond 'Free'", AEA Papers and Proceedings, 108 : 38–42.DOI: 10.1257/pandp.20181003, https://www.aeaweb.org/conference/2018/preliminary/paper/2Y7N88na

[Jain2020] Tushar Jain, "Exploring the Design Space of Liquidity Mining", Multicoin Capital Blog, Aug 13, 2020, https://multicoin.capital/2020/08/13/exploring-the-design-space-of-liquidity-mining/

[JSON2020] Douglas Crockford, "Introducing JSON", Webpage, last accessed Aug 19, 2020, https://www.json.org/

[KellyCriterion2017] "Kelly Criterion", Wikipedia, last accessed Feb. 17, 2018, https://en.wikipedia.org/wiki/Kelly_criterion

[Kleros2020] "Kleros", homepage, last accessed Aug. 20, 2020, https://kleros.io/

[Kuhn2019a] Erwin Kuhn, "Let's Talk About Data Pricing — Part I", Aug. 13, 2019, https://blog.oceanprotocol.com/lets-talk-about-data-pricing-part-i-bbc9cf781d9f

[Kuhn2019b] Erwin Kuhn, "Let's Talk About Data Pricing — Part II", Aug. 21, 2019, https://blog.oceanprotocol.com/value-of-data-part-two-pricing-bc6c5127e338

[Lao2019] Zefram Lao, "DAOmesh: Large-scale collaboration through a network of small DAOs connected by bounties", blog, May 17, 2019, https://www.zeframlou.com/2019/05/daomesh-large-scale-collaboration.html

[Lee2020] Andrew Lee, "Announcing Karma DAO: First-Ever Token-Permissioned Networking Chat Group on Telegram", July 20, 2020, https://medium.com/@andrwlee/announcing-karma-dao-first-ever-token-permissioned-networking-chat-group-on-telegram-5feab7a54def

[Lockyer2018] Matt Lockyer, "ERC-998 Composable Non-Fungible Token Standard", Apr. 15, 2018, https://github.com/ethereum/eips/issues/998

[Martinelli2020] Fernando Martinelli, "Balancer Liquidity Mining Begins", Balancer Blog, May 29, 2020, https://medium.com/balancer-protocol/balancer-liquidity-mining-begins-6e65932eaea9

[McConaghy2016] Trent McConaghy, "The DCS Triangle: Decentralized, Consistent, Scalable", July 10, 2016, https://blog.bigchaindb.com/the-dcs-triangle-5ce0e9e0f1dc

[McConaghy2018] Trent McConaghy, "Mission & Values for Ocean Protocol", July 23, 2018, https://blog.oceanprotocol.com/mission-values-for-ocean-protocol-aba998e95b8

[McConaghy2019a] Trent McConaghy, "Ocean on PoA vs. Ethereum Mainnet?", Ocean Protocol Blog, Feb. 12, 2019, https://blog.oceanprotocol.com/ocean-on-poa-vs-ethereum-mainnet-decd0ac72c97

[McConaghy2019b] Trent McConaghy, "Datatokens 1: Data Custody", Ocean Protocol Blog, FNov. 8, 2019, https://blog.oceanprotocol.com/data-tokens-1-data-custody-1d0d5ae66d0c

[McConaghy2019c] Trent McConaghy, "Datatokens 2: Non-Fungible, Fungible and Composable Datatokens", Ocean Protocol Blog, Nov. 25, 2019, https://blog.oceanprotocol.com/data-tokens-2-fungible-composable-54b6e0d28293

[McConaghy2019d] Trent McConaghy, "Datatokens 3: Data and Decentralized Finance (Data * DeFi)", Ocean Protocol Blog, Dec. 3, 2019, https://blog.oceanprotocol.com/data-tokens-3-data-and-decentralized-finance-data-defi-d5c9a6e578b7

[McConaghy2020a] Trent McConaghy, "Radical Markets and the Data Economy", Ocean Protocol Blog, Mar. 5, 2020, https://blog.oceanprotocol.com/radical-markets-and-the-data-economy-4847c272f5

[McConaghy2020b] Trent McConaghy, "Web3 Sustainability Survey: A Review of Ecosystem Funding Programs", Ocean Protocol Blog, Mar. 14, 2020, https://blog.oceanprotocol.com/web3-sustainability-i-survey-of-ecosystem-funding-programs-ffa2bb235df5

[McConaghy2020c] Trent McConaghy, "How Does Ocean Compute-to-Data Related to Other Privacy-Preserving Approaches?", Ocean Protocol Blog, May 28, 2020, https://blog.oceanprotocol.com/how-ocean-compute-to-data-relates-to-other-privacy-preserving-technology-b4e1c330483

[McConaghy2020d] Trent McConaghy, "The Web3 Sustainability Loop. A system design for long-term growth of Web3 projects, with application to Ocean Protocol", Ocean Protocol Blog, Sep. 1, 2020, https://blog.oceanprotocol.com/the-web3-sustainability-loop-b2a4097a36e

[McConaghy2020e] Trent McConaghy, "Ocean Datatokens: From Money Legos to Data Legos. How DeFi Helps Data, and Data Helps DeFi", Ocean Protocol Blog, Sep. 8, 2020, https://blog.oceanprotocol.com/ocean-datatokens-from-money-legos-to-data-legos-4f867cec1837

[McConaghy2020f] Trent McConaghy, "Ocean Market: An Open-Source Community Marketplace for Data. Featuring OCEAN Staking, Automated Market Makers, and Initial Data Offerings," Ocean Protocol Blog, Sep. 24, 2020, https://blog.oceanprotocol.com/ocean-market-an-open-source-community-marketplace-for-data-4b99bedacdc3

[McConaghy2020g] Trent McConaghy, "Ocean Token Model: System Dynamics for Near-Term Growth and Long-Term Sustainability," Ocean Protocol Blog, Oct. 2, 2020, https://blog.oceanprotocol.com/ocean-token-model-3e4e7af210f9

[McConaghy2020h] Trent McConaghy, "Ocean Protocol V3 Architecture Overview: Simplicity and Interoperability via a Datatokens Core," Ocean Protocol Blog, Oct. 12, 2020, https://blog.oceanprotocol.com/ocean-protocol-v3-architecture-overview-9f2fab60f9a7

[McConaghy2020i] Trent McConaghy, "On Selling Data in Ocean Market: Selling Your Proprietary Data, Others' Proprietary Data, and Value-Adds to Open Data," Ocean Protocol Blog, Oct. 19, 2020, https://blog.oceanprotocol.com/on-selling-data-in-ocean-market-9afcfa1e6e43

[McConaghy2020j] Trent McConaghy, "On Staking on Data in Ocean Market: Earning Opportunities — and Risks — in Ocean's Community Marketplace," Ocean Protocol Blog, Oct. 22, 2020, https://blog.oceanprotocol.com/on-staking-on-data-in-ocean-market-3d8e09eb0a13

[McDonald2020] Mike McDonald, "Building Liquidity into Token Distribution", Mar. 4, 2020, https://medium.com/balancer-protocol/building-liquidity-into-token-distribution-a49d4286e0d4

[Melon2020] "Melon Protocol", homepage, last accessed Aug. 19, 2020, https://melonprotocol.com/

[MOBI2020] “MOBI - Mobility Open Blockchain Initiative”. Homepage. Last accessed Aug. 20, 2020, https://dlt.mobi/

[Monegro2020] Joel Monegro, “Proof of Liquidity”, Placeholder blog, May 22, 2020, https://www.placeholder.vc/blog/2020/5/22/proof-of-liquidity

[Napheys2020] Alex Napheys, “Introducing OceanDAO: A grants DAO curated by the Ocean community, for growth & long-term sustainability”, Ocean Protocol Blog, Nov. 30, 2020, https://blog.oceanprotocol.com/introducing-oceandao-efe9ed681b84

[OceanBlog2020a] Ocean Protocol Team, “dexFreight and Ocean Protocol Partner to Enable Transportation and Logistics Companies To Monetize Data”, Feb. 13, 2020, https://blog.oceanprotocol.com/dexfreight-ocean-protocol-partner-to-enable-transportation-logistics-companies-to-monetize-data-7aa839195ac

[OceanBlog2020b] Ocean Protocol Team, “Ocean Protocol delivers Proof-of-Concept for Daimler AG in collaboration with Daimler South East Asia”, July 7, 2020, https://blog.oceanprotocol.com/ocean-protocol-delivers-proof-of-concept-for-daimler-ag-in-collaboration-with-daimler-south-east-564aa7d959ca

[OceanBlog_Roadmap2019] “Ocean Roadmap || Setting sail to 2021”, Ocean Protocol Blog, Feb. 13, 2019, https://blog.oceanprotocol.com/ocean-roadmap-setting-sail-to-2021-70c2545547a7

[OceanBlog_Roadmap2020] “Ocean Product Update || 2020”, Ocean Protocol Blog, Mar .19, 2020, https://blog.oceanprotocol.com/ocean-product-update-2020-f3ae281806dc

[OceanContracts2020] “Ocean Contracts”, Ocean Protocol GitHub, last accessed Aug. 31, 2020, https://github.com/oceanprotocol/ocean-contracts

[OceanDocs2020] “Ocean Protocol Software Documentation”, last accessed Aug 20, 2020, https://docs.oceanprotocol.com/

[OceanDAOwiki] “OceanDAO Wiki”, homepage, last accessed Dec 9, 2020, https://github.com/oceanprotocol/oceandao/wiki

[OceanLibJs2020] “Ocean-lib: JavaScript”, Ocean Protocol GitHub, last accessed Aug. 31, 2020, https://github.com/oceanprotocol/ocean-lib-js

[OceanLibPy2020] “Ocean-lib: Python”, Ocean Protocol GitHub, last accessed Aug. 31, 2020, https://github.com/oceanprotocol/ocean-lib-py

[OceanMarket2020] “Ocean Market”, Ocean Protocol GitHub, last accessed Aug. 31, 2020, https://github.com/oceanprotocol/market

[OceanReact2020] “Ocean React Hooks”, Ocean Protocol GitHub, last accessed Aug. 31, 2020, https://github.com/oceanprotocol/react

[Ogundeji2016] Olusegin Ogundeji, “Antonopoulos: Your Keys, Your Bitcoin. Not Your Keys, Not Your Bitcoin”, CoinTelegraph, Aug 10, 2016, https://cointelegraph.com/news/antonopoulos-your-keys-your-bitcoin-not-your-keys-not-your-bitcoin

[OEP7_2019] “Ocean Enhancement Proposal 7: Decentralized Identifiers,” last accessed Feb. 11, 2019, https://github.com/oceanprotocol/OEPs/tree/master/7

[OEP8_2019] “Ocean Enhancement Proposal 8: Assets Metadata Ontology,” last accessed Feb. 11, 2019, https://github.com/oceanprotocol/OEPs/tree/master/8

[OpenMined2020] “OpenMined”, homepage, last accessed Aug. 20, 2020, https://www.openmined.org/

[Oauth2020] Oauth Team, “Oauth 2.0 Spec”, Oauth Website, last accessed Aug. 19, 2020, https://oauth.net/2/

[Parity2020] “Parity Substrate: Build your own blockchains”, last accessed Aug. 19, 2020, https://www.parity.io/substrate/

[Patel2020] Manan Patel, “Technical Guide to Ocean Compute-to-Data”, Ocean Protocol Blog, May 19, 2020, https://blog.oceanprotocol.com/v2-ocean-compute-to-data-guide-9a3491034b64

[Pregelj2017] Jernej Pregelj, “CRAB - Create. Retrieve. Append. Burn”, BigchainDB blog, Oct. 19, 2017, https://blog.bigchaindb.com/crab-create-retrieve-append-burn-b9f6d111f460?gi=209a8d51b780

[Rouviere2017] Simon de la Rouviere, “Introducing Curation Markets: Trade Popularity of Memes & Information (with code)!”, Medium, May 22, 2017, https://medium.com/@simondlr/introducing-curation-markets-trade-popularity-of-memes-information-with-code-70bf6fed9881

[Samani2018] Kyle Samani, “New Models for Utility Tokens”, Feb. 13, 2018, https://multicoin.capital/2018/02/13/new-models-utility-tokens/

[SchemaOrg2020] “Schema.org DataSet schema”, last accessed Feb. 11, 2019, https://schema.org/Dataset

[Solidity2020] “Solidity”, Wikipedia, last accessed Aug. 21, 2020, https://en.wikipedia.org/wiki/Solidity

[Set2020] “TokenSets”, homepage, last accessed Aug. 19, 2020, https://www.tokensets.com/

[Shopify2020] “Shopify”, homepage, last accessed Aug. 21, 2020, https://shopify.com

[Simler2018] Kevin Simler and Robin Hanson, "The Elephant in the Brain: Hidden Motives in Everyday Life", Oxford University Press, Jan. 2018, https://www.amazon.com/Elephant-Brain-Hidden-Motives-Everyday/dp/0190495995

[Simmons2017] Andrew Simmons, "Prediction markets: How betting markets can be used to create a robust AI", Sept. 13, 2017, http://dstil.ghost.io/prediction-markets/amp/

[Stehlik2019] Philip Stehlik, "Centrifuge's Dev Fund DAO - The Lab", Centrifuge Blog, Nov. 1, 2019, https://medium.com/centrifuge/centrifuges-dev-fund-dao-the-lab-7116fce19c43

[SWP2020] "Saskatchewan Wheat Pool", Wikipedia, last accessed Aug. 21, 2020, https://en.wikipedia.org/wiki/Saskatchewan_Wheat_Pool

[TokenBridge2020] "About the AMB", TokenBridge homepage, last accessed Sept. 3, 2020, https://docs.tokenbridge.net/amb-bridge/about-amb-bridge

[TokenSPICE2020] "TokenSPICE: Token simulator using agent-based modeling", Ocean Protocol GitHub organization, open-sourced in Nov 2020, https://github.com/oceanprotocol/tokenspice

[Turley2020] Cooper Turley, "Moloch Evolved: V2 Primer", Mar. 27, 2020, https://medium.com/raid-guild/moloch-evolved-v2-primer-25c9cdeab455

[Uma2020] Uma Team, "UMA Initial Uniswap Listing", Apr. 22, 2020, https://medium.com/uma-project/umas-initial-uniswap-listing-afa7b6f6a330

[Unisocks2020] "Unisocks", homepage, https://unisocks.exchange/

[Uniswap2020] Uniswap Team, "Uniswap V2 Overview", Uniswap Blog, Mar. 23, 2020, https://uniswap.io/blog/uniswap-v2/

[Vogelsteller2015] Fabian Vogelsteller and Vitalik Buterin, "ERC-20 Token Standard", Nov. 19, 2015, https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md

[Vogelsteller2020] Fabian Vogelsteller, "rICO - The Reversible ICO", Lukso Blog, Apr. 13, 2020, https://medium.com/lukso/rico-the-reversible-ico-5392bf64318b

[Warwick2020] Kain Warwick, Twitter, June 16, 2020, https://twitter.com/kaiynne/status/1272875216931131398

[W3C2019] W3C, "Verifiable Credentials Data Model 1.0", W3C Recommendation, November 2019, https://www.w3.org/TR/vc-data-model/

[Xdai2020a] Xdai Team, "Xdai Chain", homepage, https://www.xdaichain.com/

[Xdai2020b] Xdai Team, "EasyStaking", xDAI STAKE documentation, Aug. 2020, https://www.xdaichain.com/for-stakers/easy-staking

[XRoad2020] "X-Road", Wikipedia, last accessed Aug. 19, 2020, https://en.wikipedia.org/wiki/X-Road

[Yearn2020] "gYFI time weighted voting power distribution", Yearn governance, last accessed Aug. 29, 2020, https://gov.yearn.finance/t/gyfi-time-weighted-voting-power-distribution/3314

Acknowledgements

Thanks very much to the following people for reviewing the paper and providing feedback: Simon de la Rouviere, Julien Thevenard, Bruce Pon, Alex Coseru, Monica Botez, Sarah Vallon, David Holtzman, Simon Mezgec, Robin Lehmann, Laurent Rochat, Gary Latta, Andreas Fauler. Thanks to my excellent colleagues at Ocean Protocol for the collaboration in building towards this. Finally, thanks to the broader Ocean community for their ongoing support. --Trent

11. Appendices

11.1. Appendix: Details of Publish / Consume

This section elaborates on how a publisher ("Alice" here) and a consumer ("Bob") interact, including the usage of a data storage service (Google Drive here), and Ocean Provider service, and Ethereum mainnet (for storing the datatoken and the metadata). Figure 8 illustrates. Now let's walk through the steps of publish and consume.

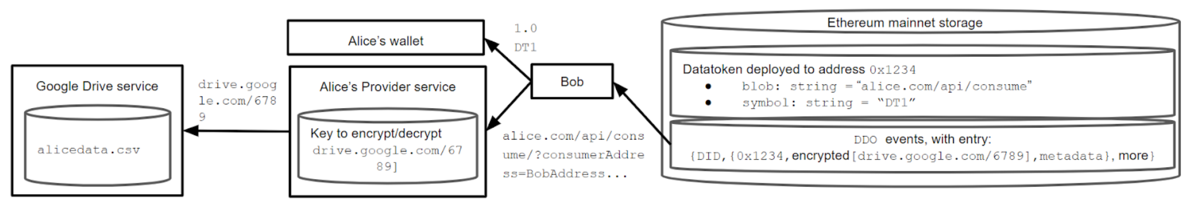

Figure 8: Details of Publish and Consume

Alice publishes:

1. Alice is running a Provider service with uri `alice.com/api/consume`. endpoint
2. Alice stores `alicedata.csv` in Google Drive, and gets a url for it: `drive.google.com/6789`
3. Alice deploys a datatoken contract to Ethereum mainnet with symbol = "`DT1`" and `blob = {"t": "1", "url": "http:example-metadata-service.com"}` (a JSON object). The Ethereum mainnet assigns the address `0x1234` to this new contract.
4. Alice constructs a metadata dict. It includes these fields: `name, dateCreated, author, license, type="dataset", files`. "`files`" is a list with one entry, a dict; that dict's "`url`" field points to the provider uri: `drive.google.com/6789`
5. Alice registers the asset on-chain, via calling `ocean_assets.create(metadata, http:alice.com/api/consume, 0x1234)`. It does the following:
    - Creates a DDO `asset` and assigns it a DID
    - Adds metadata to `asset`. Adds datatoken address `0x1234` to asset
    - Encrypts file url `drive.google.com/6789` and adds it to asset, at the same time removes the file url from the metadata
    - Adds a download service to asset which has the provider uri: http://alice.com/api/consume
    - Stores asset into Ethereum via DDO.create() which emits an event of `{DID,{0x1234, encrypted[drive.google.com/6789], metadata}, more}`. This means Alice doesn't need to run her own metadata store.

Alice mints some DT1 tokens, and Bob acquires 1.0 DT1.

Bob consumes:

1. Bob unlocks access, by a call to smart contract `startOrder()`, filling the arguments with info found in DDOContract entry for DT1. This call will send 1.0 DT1 to Alice's Eth wallet. It returns a tx id `0x9876`.
2. Bob calls Alice's provider service `alice.com/api/consume?consumerAddress=BobAddress?dataToken=0x1234?transferTxId=0x9876`. He has to do this before the timeout (e.g. 1h). If his connection drops, he can try again.

11.2. Appendix: The Datatoken Minting Attack, and Resolution

Consider the following. A publisher deploys a new datatoken contract DTX, and a DTX-OCEAN pool. Then others provide more liquidity. Then the publisher mints a *massive* number of DTX (if not

hardcoded at creation), adds those DTX to the pool, which then allows them to basically drain all the OCEAN from the pool. Or, they sell the DTX into the pool in return for OCEAN. We call this the Datatoken Minting Attack. It's an AMM “rug pull” for datatoken pools.

We considered several approaches to address this. One was to ensure that when DTX liquidity is added, at least that amount must be added in OCEAN as well. However, this does not prevent the publisher from selling down datatokens directly. Another approach was to ensure that the publisher mints all DTX into the pool immediately; but this means they have to provide a lot of OCEAN immediately too. We considered other approaches too. However, they all added complexity, and ways for attackers to circumvent by simply not using Ocean defaults. We expect better solutions in the future, as the problem is general ("rug pulls").

We decided to approach this in the same way that token projects do in the broader blockchain ecosystem: a social contract. Specifically, there are *no* constraints on how liquidity is added or removed. The DTX publisher has an implicit social contract with buyers and LPs of DTX; if the DTX publisher does end up performing a minting attack then buyers and LPs will steer clear of that publisher in the future. The GUI provides some warnings to buyers and LPs.

There’s a complementary approach: make DTX token also a bonding curve (BC) [Rouviere2017]. In this BC, OCEAN must be staked to mint more DTX tokens, typically at an increasing price. DTX tokens can be burned to receive back OCEAN. The BC is now the primary market for DTX, and AMMs etc become the secondary markets [Unisocks2020][Lee2020]. This approach is complementary rather than a full solution, because it provides tighter rails on the price of DTX compared to more open-ended primary markets. We plan to implement this in a release after the V3.0 release.