SNI Hackathon
An Intro to Ocean Protocol

Trent McConaghy

ocean

Slides: http://trent.st/content/sni.pdf

oceanprotocol.com
@trentmc0

Jan 25, 2022

Outline

- Intro to Ocean
    - Basics
    - Ocean Market
    - Ocean Ecosystem
- Opportunities for SNI hackers

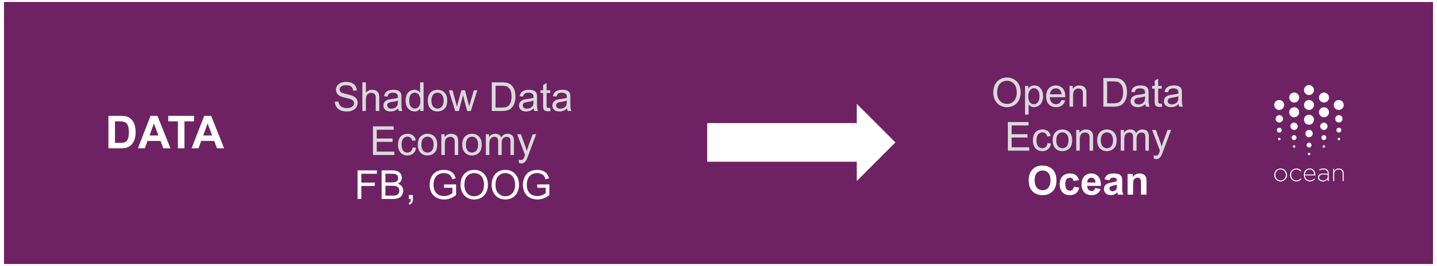
Intro to Ocean: Basics

Ocean Mission: open up data, while maintaining privacy

Towards self-sovereign data for individuals and groups.

Opaque, Power Concentrated

Transparent, Permissionless



DATA

Shadow Data Economy
FB, GOOG

Open Data Economy
Ocean

Data is a $400B industry in Europe alone

How: Use crypto tools to ***make data an asset***

What is Ocean Protocol?

Ocean is...

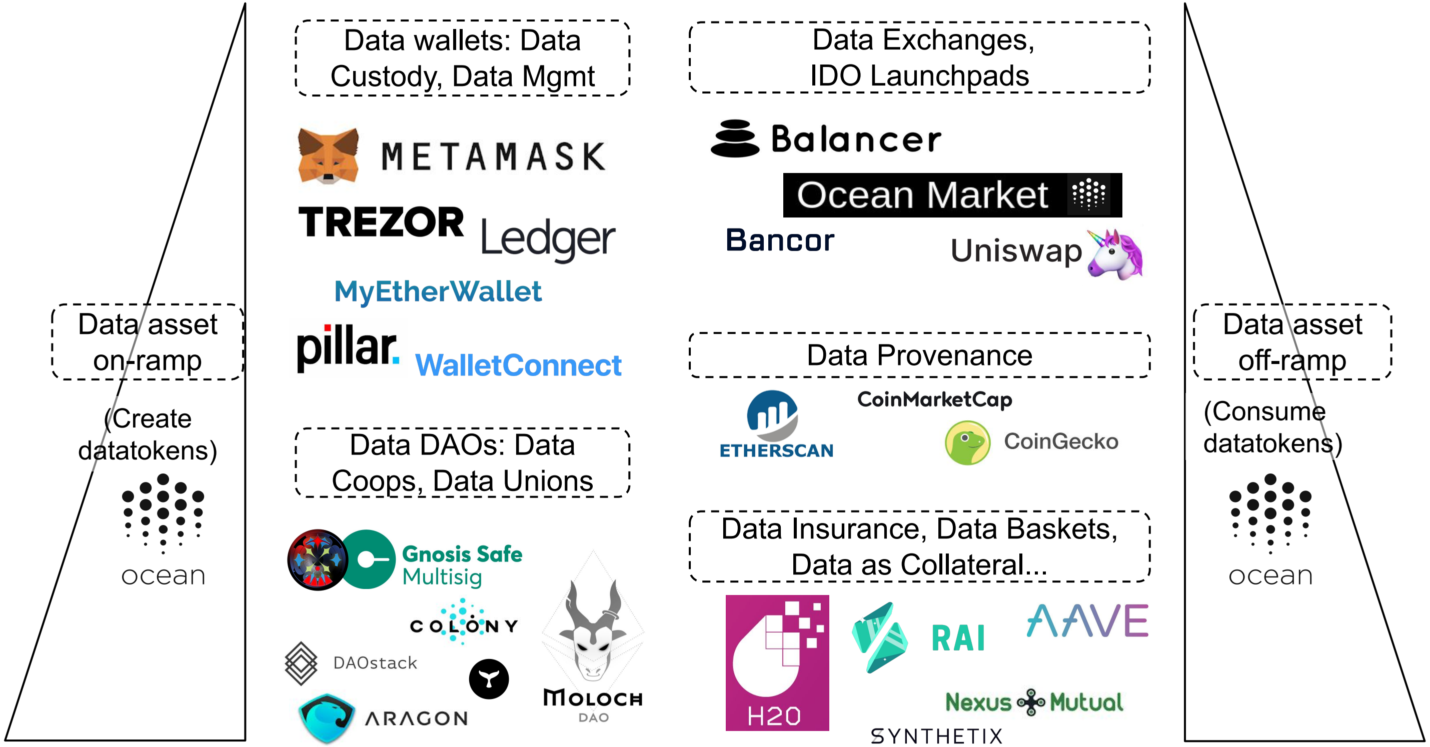
1. **A community / ecosystem** of individuals and orgs driving to the mission (initiated by Ocean Protocol Foundation)
2. **A set of tools** as public infrastructure to facilitate the mission
3. **A token** (OCEAN) with incentives to grow & sustain the ecosystem

Ocean Tools

- **Key principle**: datatokens
- **Ocean backend**: smart contracts + Py + JS. On Eth, Polygon, ..
- **Ocean Market:** dapp
- **OceanDAO:** grants program

Key principle: Ocean Datatokens

Ocean makes it easy to publish data services (deploy and mint ERC20 datatokens), and to consume data services (spend datatokens). Crypto wallets, exchanges, and DAOs become *data* wallets, exchanges, and DAOs.



Data asset on-ramp

(Create datatokens)

Data wallets: Data Custody, Data Mgmt

Data DAOs: Data Coops, Data Unions

Data Exchanges, IDO Launchpads

Data Provenance

Data Insurance, Data Baskets, Data as Collateral...

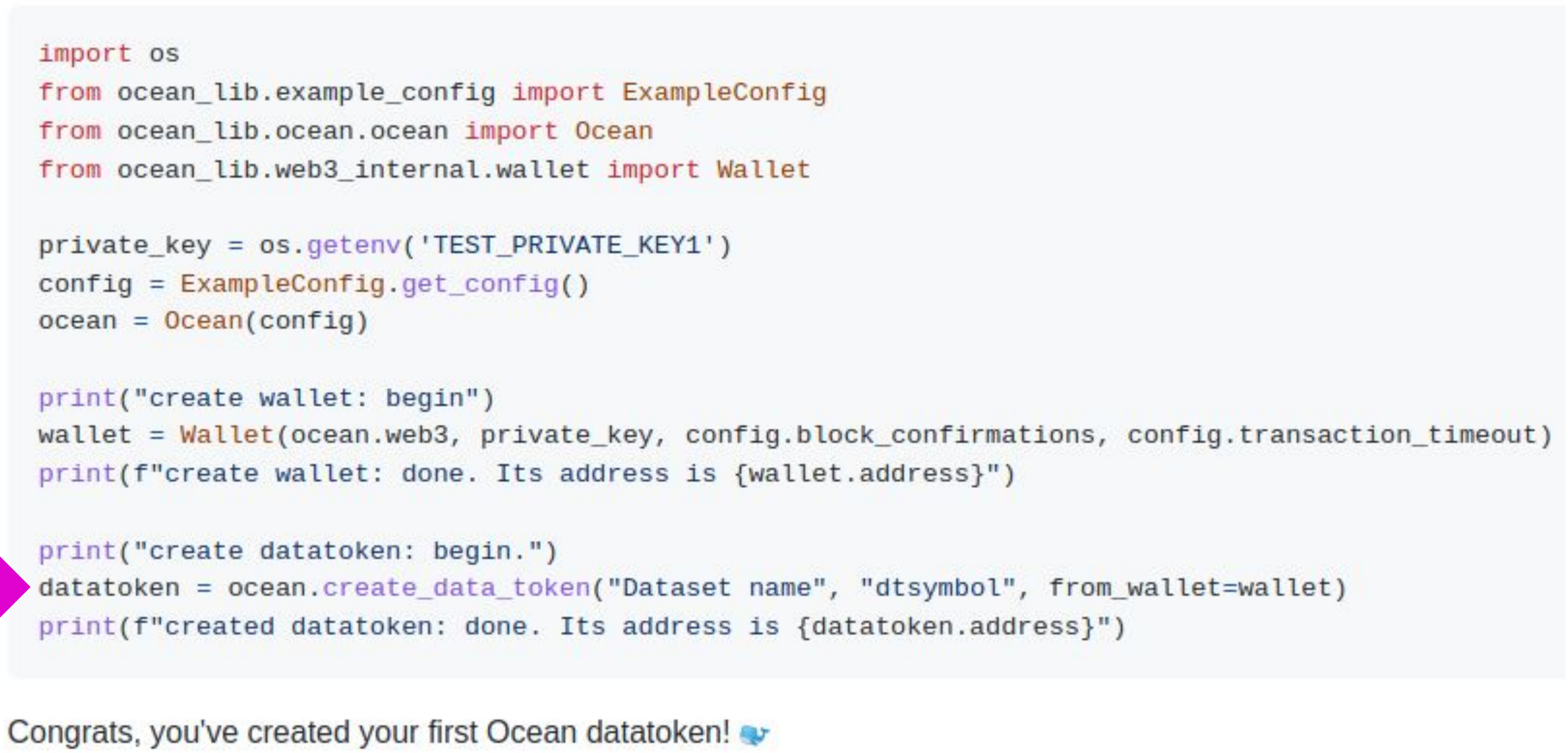
Data asset off-ramp

(Consume datatokens)

Ocean.py, js

Programmatically publish datatokens, data DEXes ..

https://github.com/oceanprotocol/ocean.py/blob/main/READMEs/datatokens-flow.md



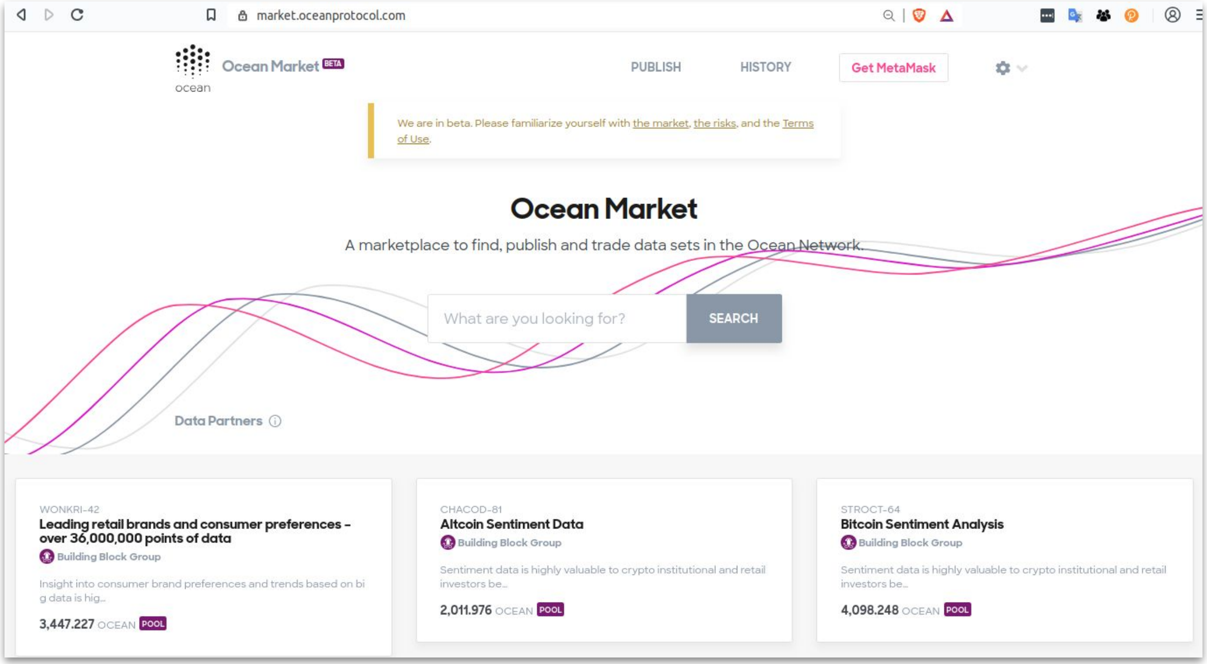
```python
import os
from ocean_lib.example_config import ExampleConfig
from ocean_lib.ocean.ocean import Ocean
from ocean_lib.web3_internal.wallet import Wallet

private_key = os.getenv('TEST_PRIVATE_KEY1')
config = ExampleConfig.get_config()
ocean = Ocean(config)

print("create wallet: begin")
wallet = Wallet(ocean.web3, private_key, config.block_confirmations, config.transaction_timeout)
print(f"create wallet: done. Its address is {wallet.address}")

print("create datatoken: begin.")
datatoken = ocean.create_data_token("Dataset name", "dtsymbol", from_wallet=wallet)
print(f"created datatoken: done. Its address is {datatoken.address}")
```

Congrats, you've created your first Ocean datatoken! 🐳

Ocean Market

market.oceanprotocol.com

- Data on-ramp (publish)
- Data off-ramp (consume)
- DEX, for data



We are in beta. Please familiarize yourself with the market, the risks, and the Terms of Use.

Ocean Market

A marketplace to find, publish and trade data sets in the Ocean Network.

Data Partners ⓘ

WONKRI-42
Leading retail brands and consumer preferences – over 36,000,000 points of data
Building Block Group
Insight into consumer brand preferences and trends based on bi g data is hig...
3,447.227 OCEAN POOL

CHACOD-81
Altcoin Sentiment Data
Building Block Group
Sentiment data is highly valuable to crypto institutional and retail investors be...
2,011.976 OCEAN POOL

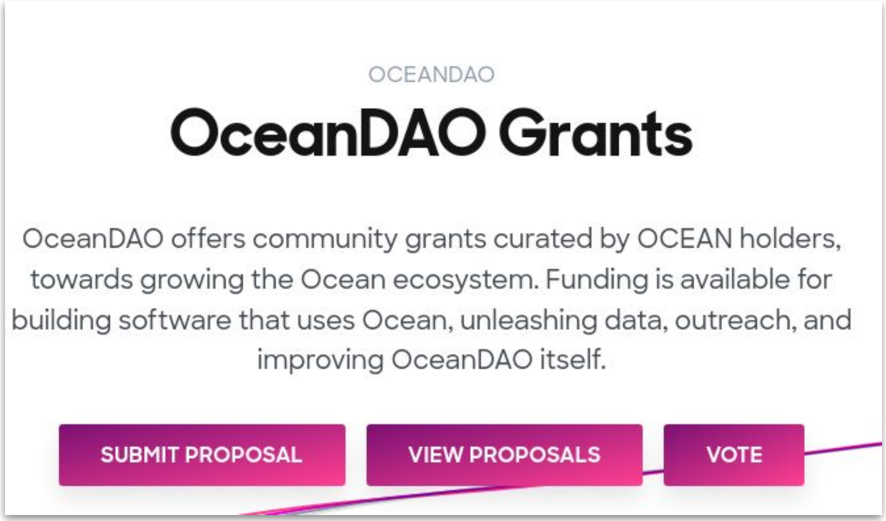
STROCT-64
Bitcoin Sentiment Analysis
Building Block Group
Sentiment data is highly valuable to crypto institutional and retail investors be...
4,098.248 OCEAN POOL

ocean

OceanDAO Grants

$100K+ available per month. Anyone can apply

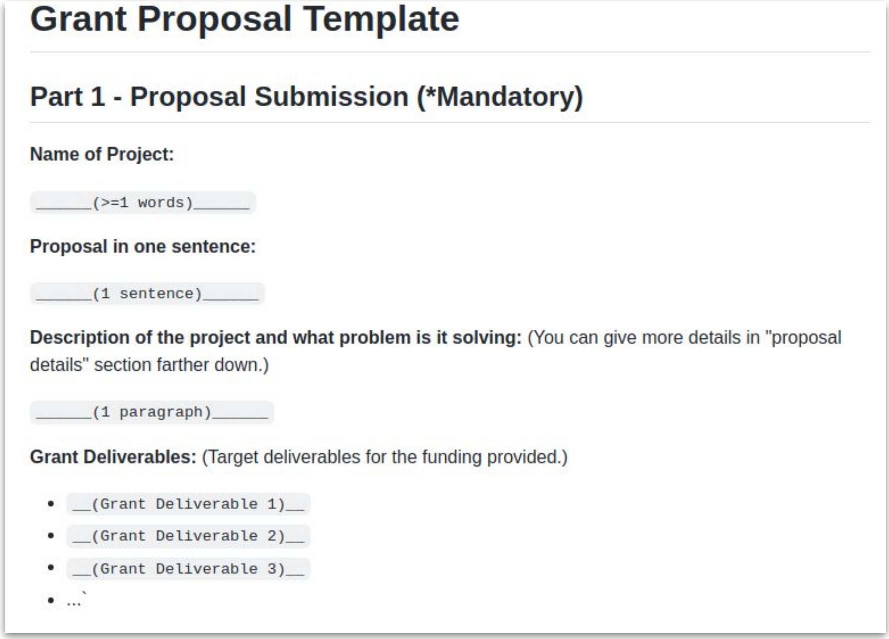
https://oceanprotocol.com/dao



OCEANDAO

OceanDAO Grants

OceanDAO offers community grants curated by OCEAN holders, towards growing the Ocean ecosystem. Funding is available for building software that uses Ocean, unleashing data, outreach, and improving OceanDAO itself.



Grant Proposal Template

Part 1 - Proposal Submission (*Mandatory)

Name of Project:

`______(>=1 words)______`

Proposal in one sentence:

`______(1 sentence)______`

Description of the project and what problem is it solving: (You can give more details in "proposal details" section farther down.)

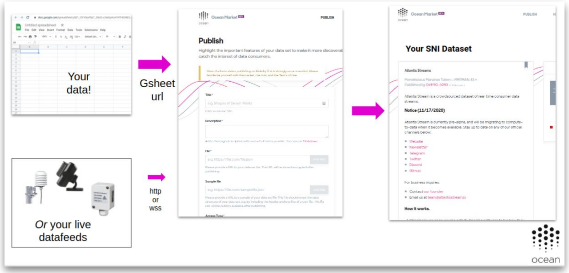
`______(1 paragraph)______`

Grant Deliverables: (Target deliverables for the funding provided.)

- `__(Grant Deliverable 1)__`
- `__(Grant Deliverable 2)__`
- `__(Grant Deliverable 3)__`
- ...`

There’s opportunity for SNI hackers

Sell climate data or ML models. Create your own data market



Your data!

Gsheet url

Or your live datafeeds

http or wss

Publish

Highlight the important features of your data set to make it more discoverab catch the interest of data consumers.

Title*

e.g. Shapes of Desert Plants

Description*

File*

e.g. https://file.com/file.json

Sample file

e.g. https://file.com/samplefile.json

Access Type*

Your SNI Dataset

Atlantis Streams

Published by 0x4f40..5083

Atlantis Stream is a crowdsourced dataset of real-time consumer data streams.

Notice (11/17/2020)

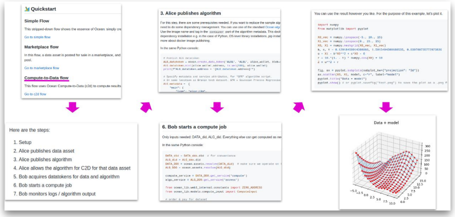
Atlantis Stream is currently pre-alpha, and will be migrating to compute-to-data when it becomes available. Stay up to date on any of our official channels below:

- Website
- Newsletter
- Telegram
- Twitter
- Discord
- Github

For business inquires:

- Contact our founder
- Email us at team@atlantisstream.io

How it works.



Quickstart

Simple Flow

This stripped-down flow shows the essence of Ocean: simply cre

Go to simple flow

Marketplace flow

In this flow, a data asset is posted for sale in a marketplace, and pool.

Go to marketplace flow

Compute-to-Data flow

This flow uses Ocean Compute-to-Data (c2d) to compute results

Go to c2d flow

Here are the steps:

1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

3. Alice publishes algorithm

For this step, there are some prerequisites needed. If you want to replace the sample alg need to do some dependency management. You can use one of the standard Ocean alg Use the image name and tag in the `container` part of the algorithm metadata. This dock dependency installation e.g. in the case of Python, OS-level library installations, pip instal more about docker image publishing.

In the same Python console:

```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob=
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regressio
ALG_metadata = {
    "main": {
        "type": "algorithm",
```

6. Bob starts a compute job

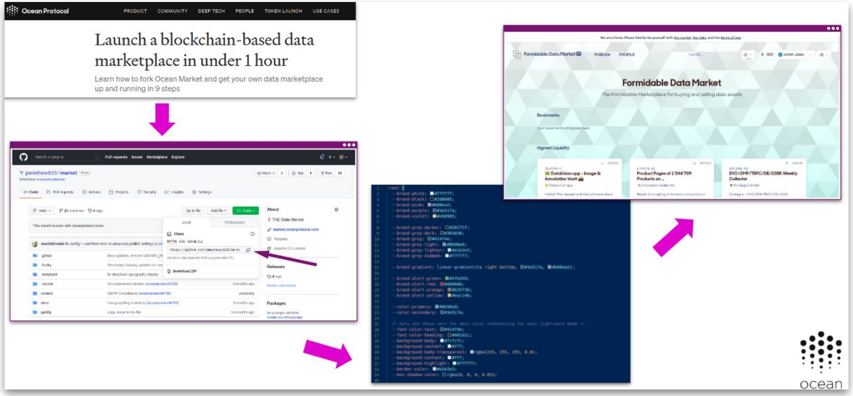
Only inputs needed: DATA_did, ALG_did. Everything else can get computed as ne

In the same Python console:

```
DATA_did = DATA_ddo.did  # for convenience
ALG_did = ALG_ddo.did
DATA_DDO = ocean.assets.resolve(DATA_did)  # make sure we operate on t
ALG_DDO = ocean.assets.resolve(ALG_did)

compute_service = DATA_DDO.get_service('compute')
algo_service = ALG_DDO.get_service('access')

from ocean_lib.web3_internal.constants import ZERO_ADDRESS
from ocean_lib.models.compute_input import ComputeInput

# order & pay for dataset
```

You can use the result however you like. For the purpose of this example, let's plot it.

```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.039788735772973836
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png f
```



Launch a blockchain-based data marketplace in under 1 hour

Learn how to fork Ocean Market and get your own data marketplace up and running in 9 steps

```
:root {
  --brand-white: #ffffff;
  --brand-black: #2d0605;
  --brand-pink: #b08ea2;
  --brand-purple: #3e517a;
  --brand-violet: #e9d985;

  --brand-grey-darker: #201f1f;
  --brand-grey-dark: #303030;
  --brand-grey: #41474e;
  --brand-grey-light: #8b98a9;
  --brand-grey-lighter: #e2e2e2;
  --brand-grey-dimmed: #f7f7f7;

  --brand-gradient: linear-gradient(to right bottom, #3e517a, #b08ea2);

  --brand-alert-green: #5fb359;
  --brand-alert-red: #d80606;
  --brand-alert-orange: #b35f36;
  --brand-alert-yellow: #eac146;

  --color-primary: #8b98a9;
  --color-secondary: #3e517a;

  /* Only use these vars for most color referencing for easy light/dark mode */
  --font-color-text: #41474e;
  --font-color-heading: #01d1c;
  --background-body: #fcfcfc;
  --background-content: #fff;
  --background-body-transparent: rgba(255, 255, 255, 0.8);
  --background-content: #fff;
  --background-highlight: #f7f7f7;
  --border-color: #e2e2e2;
  --box-shadow-color: rgba(0, 0, 0, 0.05);
```

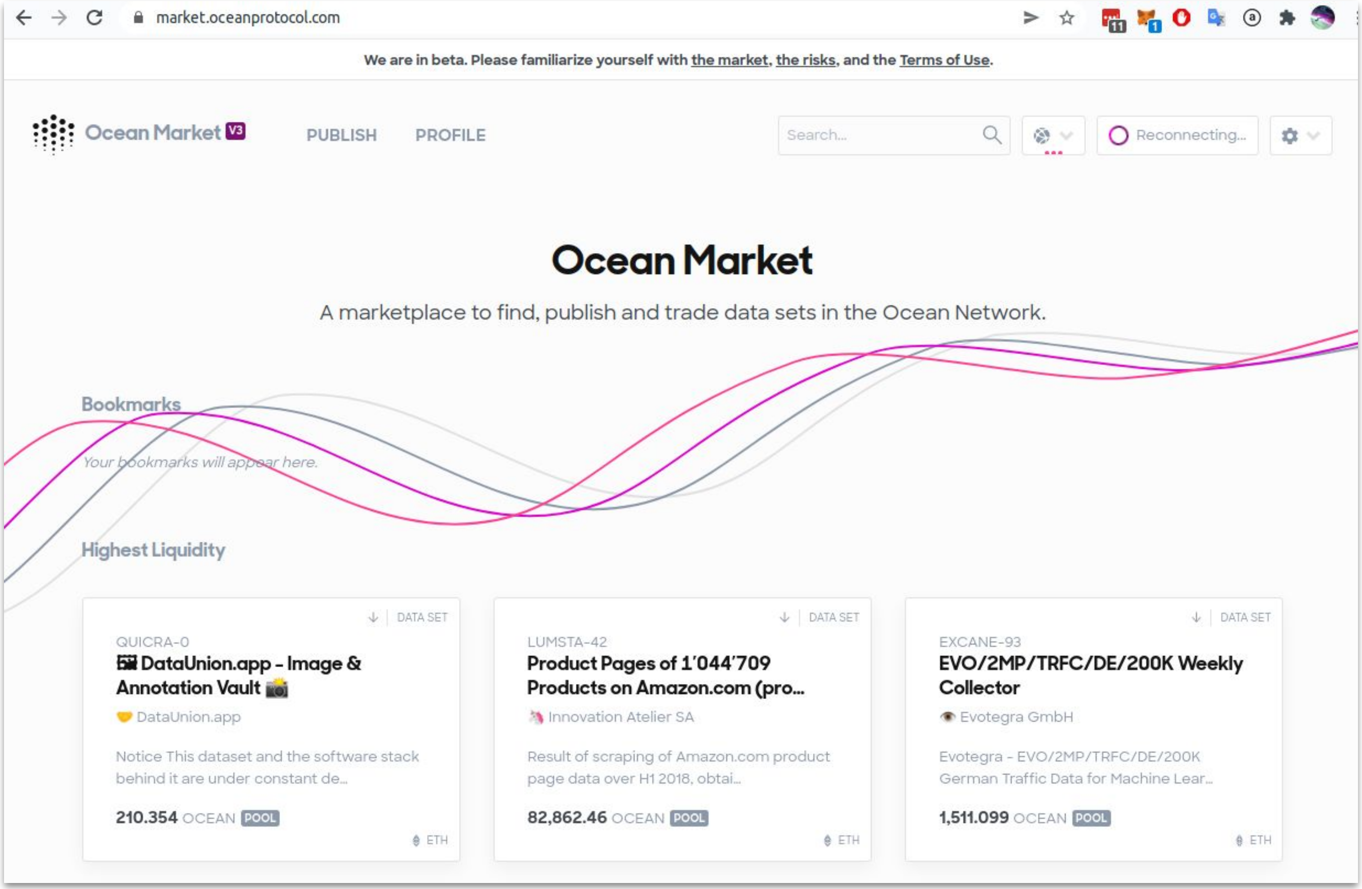
Formidable Data Market

The Formidable Marketplace for buying and selling data assets

Formidable Data Market
PUBLISH
PROFILE
Bookmarks
Highest Liquidity
DataUnion.app - Image & Annotation Vault
Product Pages of 1'044'709 Products on ...
EVO/2MP/TRFC/DE/200K Weekly Collector
ocean

Intro to Ocean: Ocean Market

Ocean Market: Splash Page



We are in beta. Please familiarize yourself with the market, the risks, and the Terms of Use.

Ocean Market

A marketplace to find, publish and trade data sets in the Ocean Network.

Bookmarks

Your bookmarks will appear here.

Highest Liquidity

DATA SET

QUICRA-0

🎞 DataUnion.app - Image & Annotation Vault 📸

DataUnion.app

Notice This dataset and the software stack behind it are under constant de...

210.354 OCEAN POOL

ETH

DATA SET

LUMSTA-42

Product Pages of 1'044'709 Products on Amazon.com (pro...

Innovation Atelier SA

Result of scraping of Amazon.com product page data over H1 2018, obtai...

82,862.46 OCEAN POOL

ETH

DATA SET

EXCANE-93

EVO/2MP/TRFC/DE/200K Weekly Collector

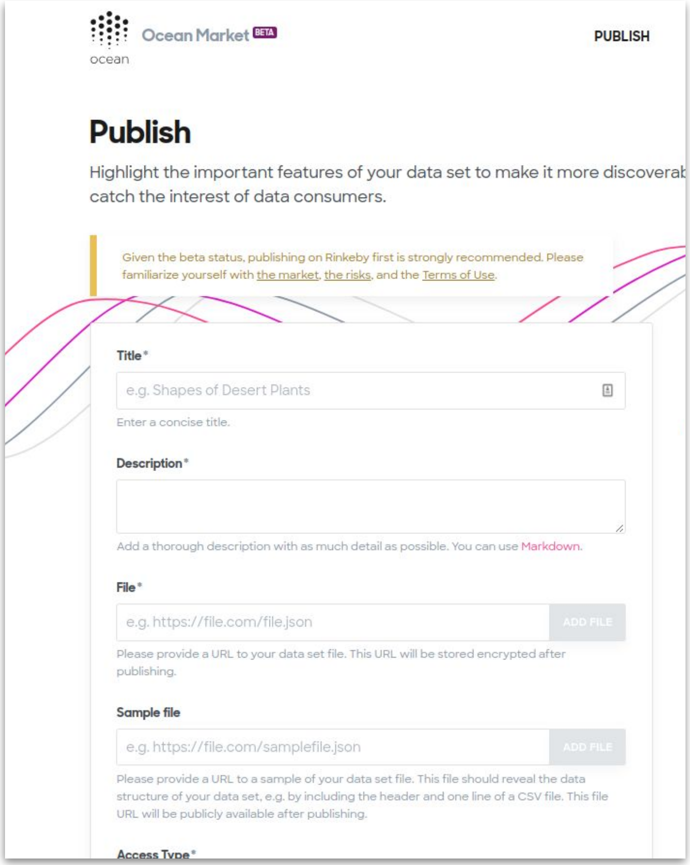
Evotegra GmbH

Evotegra - EVO/2MP/TRFC/DE/200K German Traffic Data for Machine Lear...

1,511.099 OCEAN POOL

ETH

Ocean Market: Publish Flow, for an Initial Data Offering



Publish

Highlight the important features of your data set to make it more discoverab catch the interest of data consumers.

Given the beta status, publishing on Rinkeby first is strongly recommended. Please familiarize yourself with the market, the risks, and the Terms of Use.

Title*

e.g. Shapes of Desert Plants

Enter a concise title.

Description*

Add a thorough description with as much detail as possible. You can use Markdown.

File*

e.g. https://file.com/file.json

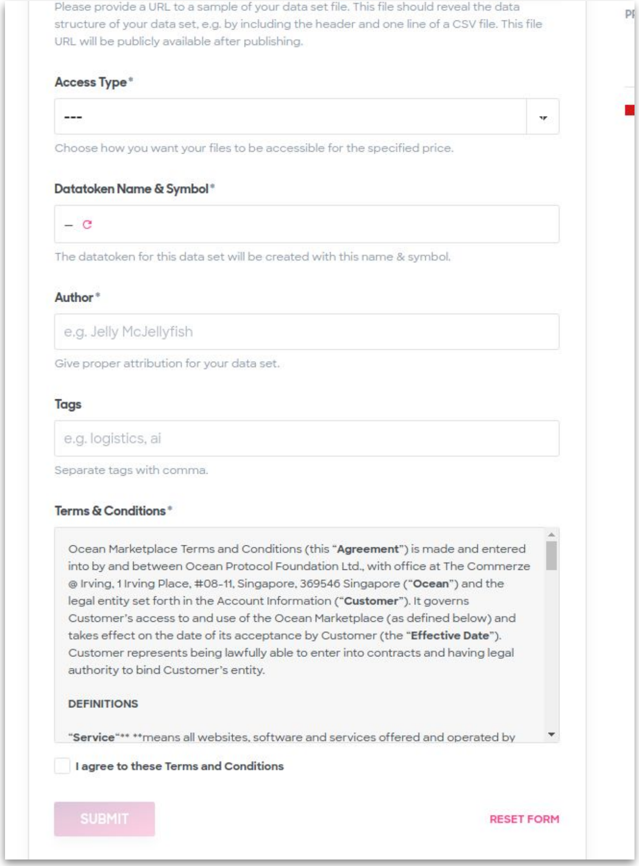
Please provide a URL to your data set file. This URL will be stored encrypted after publishing.

Sample file

e.g. https://file.com/samplefile.json

Please provide a URL to a sample of your data set file. This file should reveal the data structure of your data set, e.g. by including the header and one line of a CSV file. This file URL will be publicly available after publishing.

Access Type*



Please provide a URL to a sample of your data set file. This file should reveal the data structure of your data set, e.g. by including the header and one line of a CSV file. This file URL will be publicly available after publishing.

Access Type*

Choose how you want your files to be accessible for the specified price.

Datatoken Name & Symbol*

– C

The datatoken for this data set will be created with this name & symbol.

Author*

e.g. Jelly McJellyfish

Give proper attribution for your data set.

Tags

e.g. logistics, ai

Separate tags with comma.

Terms & Conditions*

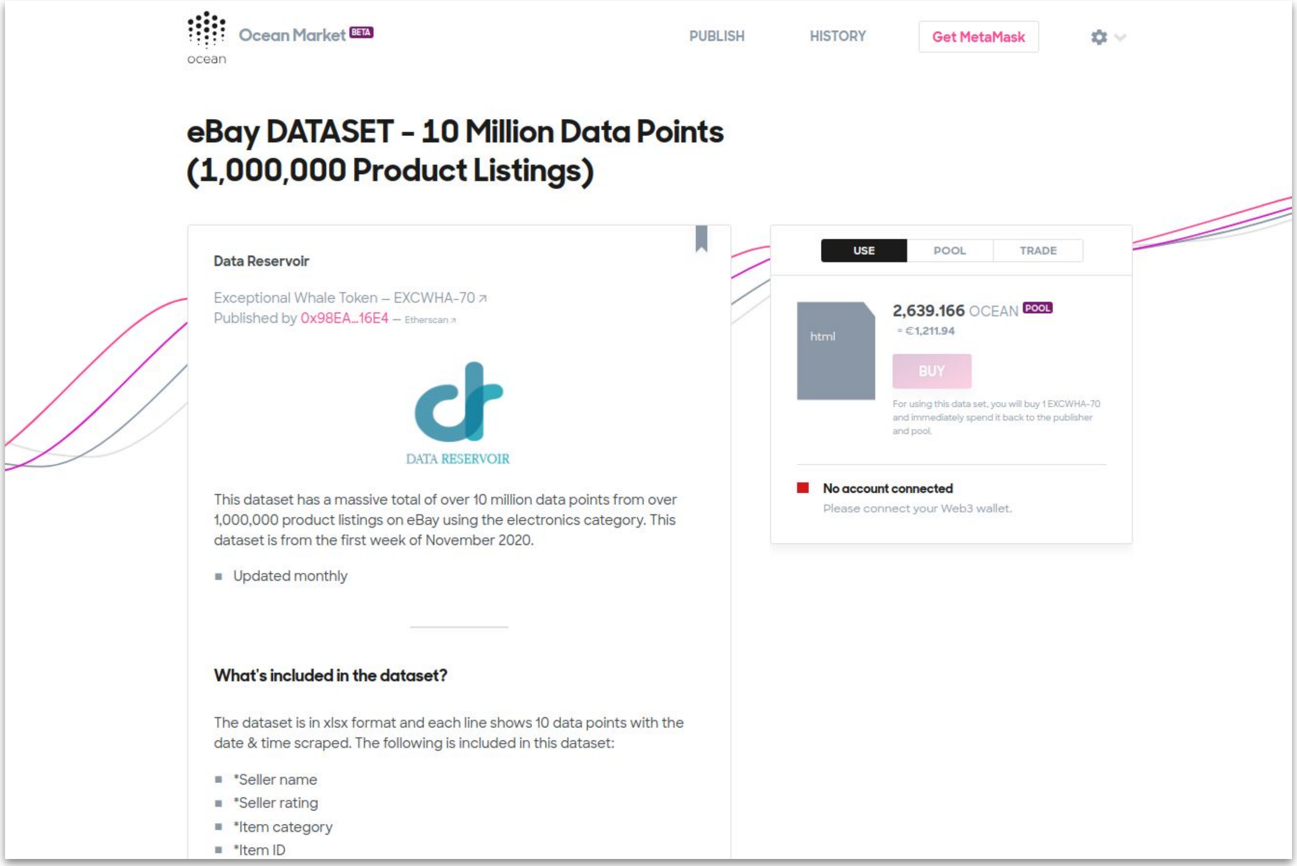
Ocean Marketplace Terms and Conditions (this "**Agreement**") is made and entered into by and between Ocean Protocol Foundation Ltd., with office at The Commerze @ Irving, 1 Irving Place, #08-11, Singapore, 369546 Singapore ("**Ocean**") and the legal entity set forth in the Account Information ("**Customer**"). It governs Customer's access to and use of the Ocean Marketplace (as defined below) and takes effect on the date of its acceptance by Customer (the "**Effective Date**"). Customer represents being lawfully able to enter into contracts and having legal authority to bind Customer's entity.

DEFINITIONS

"**Service**"** **means all websites, software and services offered and operated by

□ **I agree to these Terms and Conditions**

Example Data Asset, for Fixed Price



eBay DATASET - 10 Million Data Points (1,000,000 Product Listings)

Data Reservoir

Exceptional Whale Token – EXCWHA-70 ↗
Published by 0x98EA…16E4 – Etherscan ↗

This dataset has a massive total of over 10 million data points from over 1,000,000 product listings on eBay using the electronics category. This dataset is from the first week of November 2020.

- Updated monthly

What's included in the dataset?

The dataset is in xlsx format and each line shows 10 data points with the date & time scraped. The following is included in this dataset:

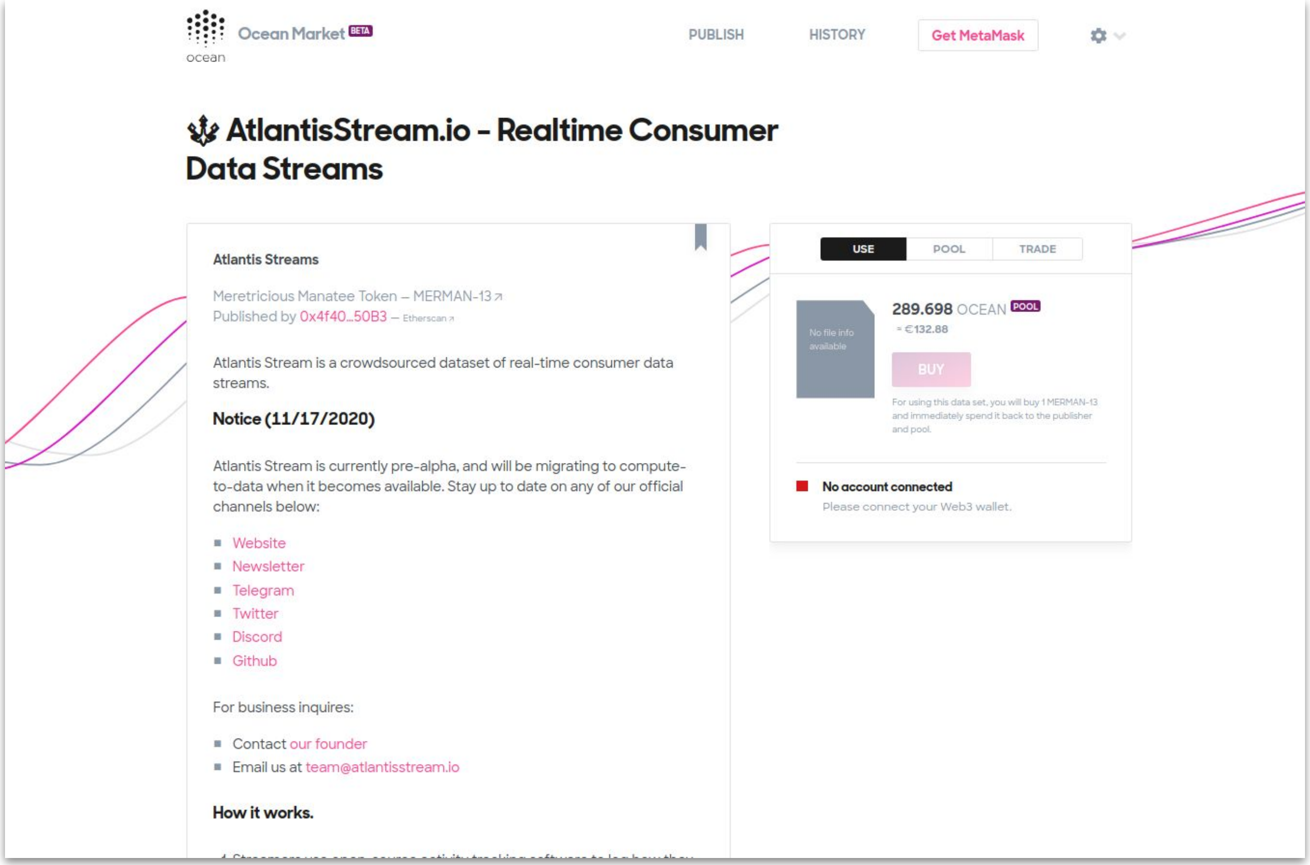
- *Seller name
- *Seller rating
- *Item category
- *Item ID

2,639.166 OCEAN POOL
≈ €1,211.94

For using this data set, you will buy 1 EXCWHA-70 and immediately spend it back to the publisher and pool.

🟥 **No account connected**
Please connect your Web3 wallet.

Example Data Asset, with Automatic Price Discovery (via AMM)



AtlantisStream.io - Realtime Consumer Data Streams

Atlantis Streams

Meretricious Manatee Token – MERMAN-13 ↗
Published by 0x4f40…50B3 – Etherscan ↗

Atlantis Stream is a crowdsourced dataset of real-time consumer data streams.

Notice (11/17/2020)

Atlantis Stream is currently pre-alpha, and will be migrating to compute-to-data when it becomes available. Stay up to date on any of our official channels below:

- Website
- Newsletter
- Telegram
- Twitter
- Discord
- Github

For business inquires:

- Contact our founder
- Email us at team@atlantisstream.io

How it works.

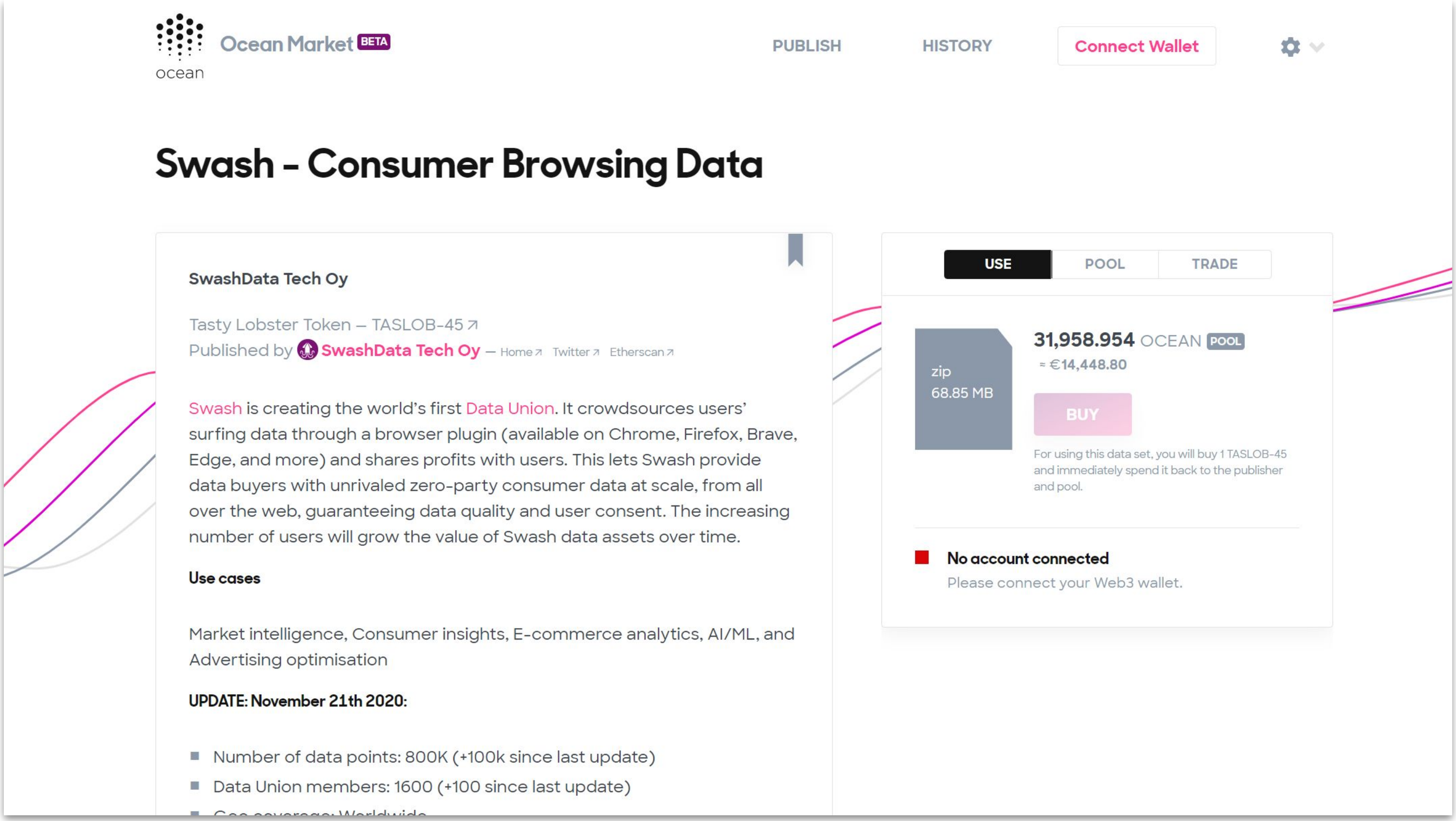
1. Streamers use open-source activity tracking software to log how they

289.698 OCEAN POOL
≈ €132.88

For using this data set, you will buy 1 MERMAN-13 and immediately spend it back to the publisher and pool.

🟥 **No account connected**
Please connect your Web3 wallet.

Example Data Asset: A Data Union



Swash – Consumer Browsing Data

SwashData Tech Oy

Tasty Lobster Token — TASLOB-45 ↗
Published by SwashData Tech Oy — Home ↗ Twitter ↗ Etherscan ↗

Swash is creating the world's first Data Union. It crowdsources users' surfing data through a browser plugin (available on Chrome, Firefox, Brave, Edge, and more) and shares profits with users. This lets Swash provide data buyers with unrivaled zero-party consumer data at scale, from all over the web, guaranteeing data quality and user consent. The increasing number of users will grow the value of Swash data assets over time.

Use cases

Market intelligence, Consumer insights, E-commerce analytics, AI/ML, and Advertising optimisation

UPDATE: November 21th 2020:

- Number of data points: 800K (+100k since last update)
- Data Union members: 1600 (+100 since last update)
- Geo coverage: Worldwide

31,958.954 OCEAN POOL
≈ €14,448.80

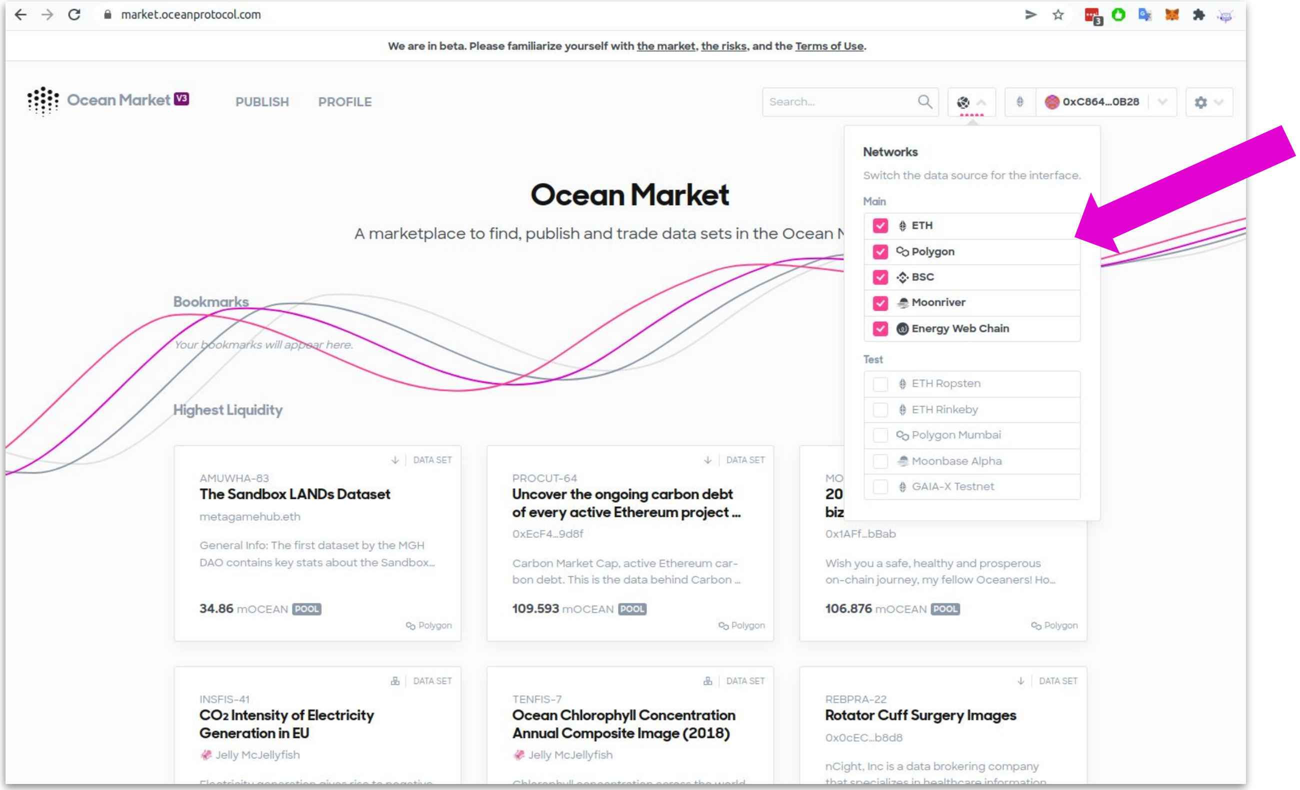
For using this data set, you will buy 1 TASLOB-45 and immediately spend it back to the publisher and pool.

🟥 **No account connected**
Please connect your Web3 wallet.

Ocean is multi-chain

Therefore low fees

blog.oceanprotocol.com/ocean-makes-multinetwork-even-simpler-c3ec6c0cbd50



market.oceanprotocol.com

We are in beta. Please familiarize yourself with the market, the risks, and the Terms of Use.

Ocean Market V3 PUBLISH PROFILE Search... 0xC864...0B28

Ocean Market

A marketplace to find, publish and trade data sets in the Ocean N

Networks

Switch the data source for the interface.

Main

- ☑ ETH
- ☑ Polygon
- ☑ BSC
- ☑ Moonriver
- ☑ Energy Web Chain

Test

- □ ETH Ropsten
- □ ETH Rinkeby
- □ Polygon Mumbai
- □ Moonbase Alpha
- □ GAIA-X Testnet

Bookmarks

Your bookmarks will appear here.

Highest Liquidity

DATA SET
AMUWHA-83
The Sandbox LANDs Dataset
metagamehub.eth
General Info: The first dataset by the MGH DAO contains key stats about the Sandbox...
34.86 mOCEAN POOL
Polygon

DATA SET
PROCUT-64
Uncover the ongoing carbon debt of every active Ethereum project ...
0xEcF4...9d8f
Carbon Market Cap, active Ethereum carbon debt. This is the data behind Carbon ...
109.593 mOCEAN POOL
Polygon

MO
20
biz
0x1AFf...bBab
Wish you a safe, healthy and prosperous on-chain journey, my fellow Oceaners! Ho...
106.876 mOCEAN POOL
Polygon

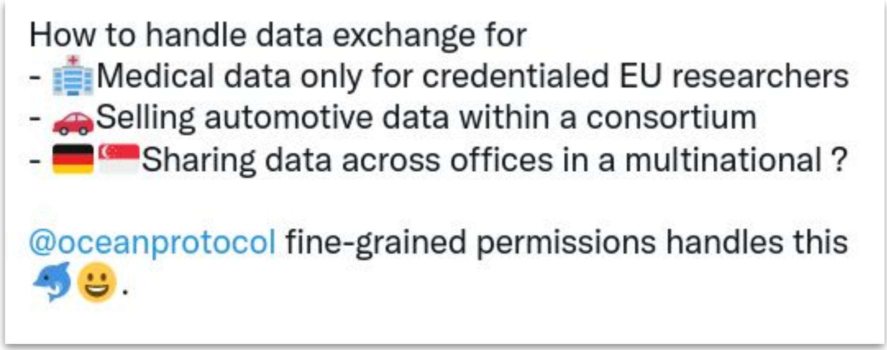
DATA SET
INSFIS-41
CO2 Intensity of Electricity Generation in EU
Jelly McJellyfish

DATA SET
TENFIS-7
Ocean Chlorophyll Concentration Annual Composite Image (2018)
Jelly McJellyfish

DATA SET
REBPRA-22
Rotator Cuff Surgery Images
0x0cEC...b8d8
nCight, Inc is a data brokering company that specializes in healthcare information

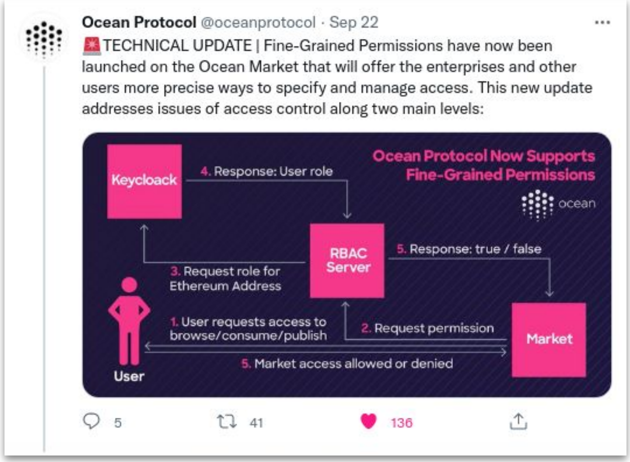
Fine-grained permissions

blog.oceanprotocol.com/fine-grained-permissions-now-supported-in-ocean-protocol-4fe434af24b9



How to handle data exchange for
- 🏥Medical data only for credentialed EU researchers
- 🚗Selling automotive data within a consortium
- 🇩🇪🇸🇬Sharing data across offices in a multinational ?

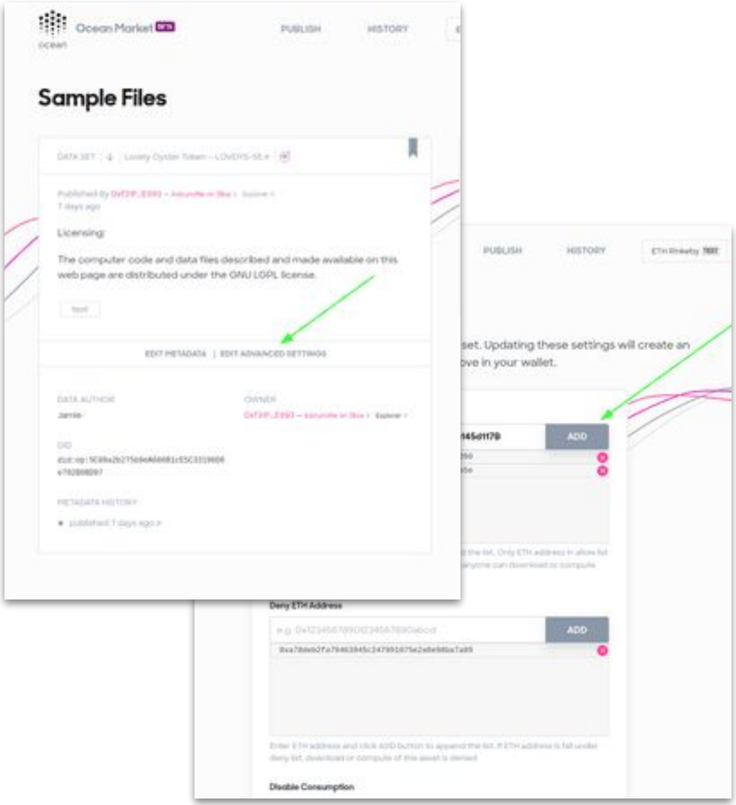
@oceanprotocol fine-grained permissions handles this 🐬😃.



Ocean Protocol @oceanprotocol · Sep 22

🚨TECHNICAL UPDATE | Fine-Grained Permissions have now been launched on the Ocean Market that will offer the enterprises and other users more precise ways to specify and manage access. This new update addresses issues of access control along two main levels:

Ocean Protocol Now Supports Fine-Grained Permissions
ocean
Keycloack
4. Response: User role
RBAC Server
5. Response: true / false
3. Request role for Ethereum Address
1. User requests access to browse/consume/publish
2. Request permission
Market
5. Market access allowed or denied
User



Sample Files

DATA SET | Lovely Oyster Token — LOVOYS-56

Published By 0xf3fF…E993 — 7 days ago

Licensing:

The computer code and data files described and made available on this web page are distributed under the GNU LGPL license.

DATA AUTHOR
Jamie

OWNER
0xf3fF…E993

DID
did:op:5C08a2b275…

METADATA HISTORY
- published 7 days ago

…set. Updating these settings will create an …ove in your wallet.

…145d117B

Deny ETH Address

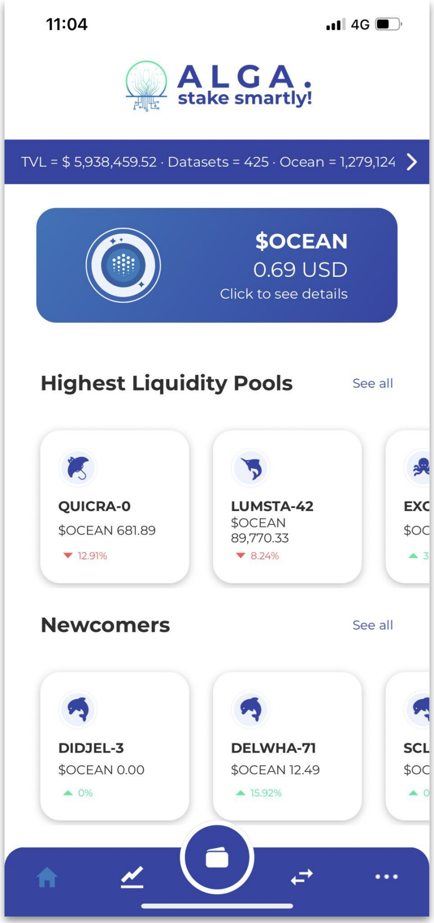
e.g. 0x1234567890123456789Oabcd

Enter ETH address and click ADD button to append the list. If ETH address is fall under deny list, download or compute of this asset is denied

Disable Consumption

Intro to Ocean: Ocean Ecosystem

Data Whale: Data Ratings, Data Wallet



TVL = $ 5,938,459.52 · Datasets = 425 · Ocean = 1,279,124

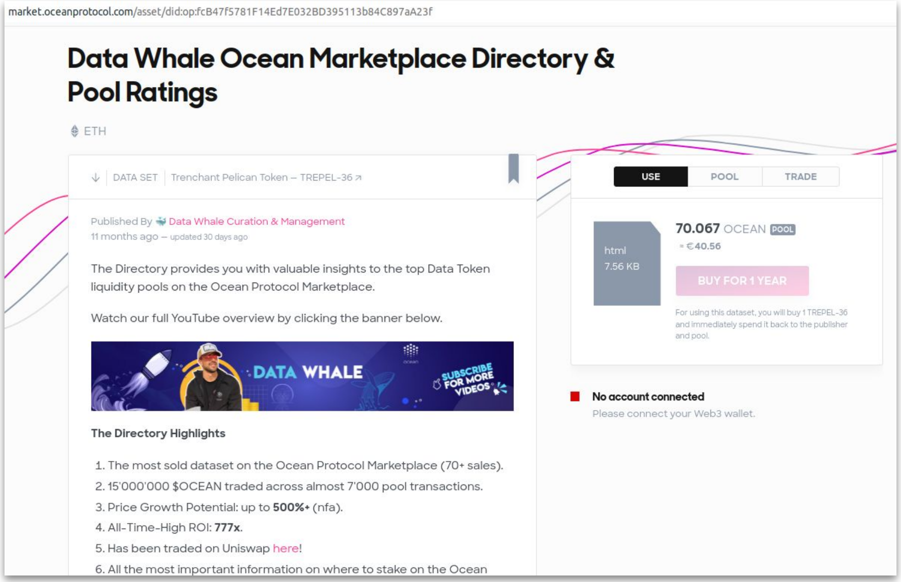
$OCEAN
0.69 USD
Click to see details

Highest Liquidity Pools — See all

QUICRA-0	LUMSTA-42	EXC
$OCEAN 681.89	$OCEAN 89,770.33	$OC
▼ 12.91%	▼ 8.24%	▲ 3

Newcomers — See all

DIDJEL-3	DELWHA-71	SCL
$OCEAN 0.00	$OCEAN 12.49	$OC
▲ 0%	▲ 15.92%	▲ 0



market.oceanprotocol.com/asset/did:op:fcB47f5781F14Ed7E032BD395113b84C897aA23f

Data Whale Ocean Marketplace Directory & Pool Ratings

ETH

DATA SET | Trenchant Pelican Token – TREPEL-36

Published By Data Whale Curation & Management
11 months ago — updated 30 days ago

The Directory provides you with valuable insights to the top Data Token liquidity pools on the Ocean Protocol Marketplace.

Watch our full YouTube overview by clicking the banner below.

The Directory Highlights

1. The most sold dataset on the Ocean Protocol Marketplace (70+ sales).
2. 15'000'000 $OCEAN traded across almost 7'000 pool transactions.
3. Price Growth Potential: up to **500%+** (nfa).
4. All-Time-High ROI: **777x**.
5. Has been traded on Uniswap here!
6. All the most important information on where to stake on the Ocean

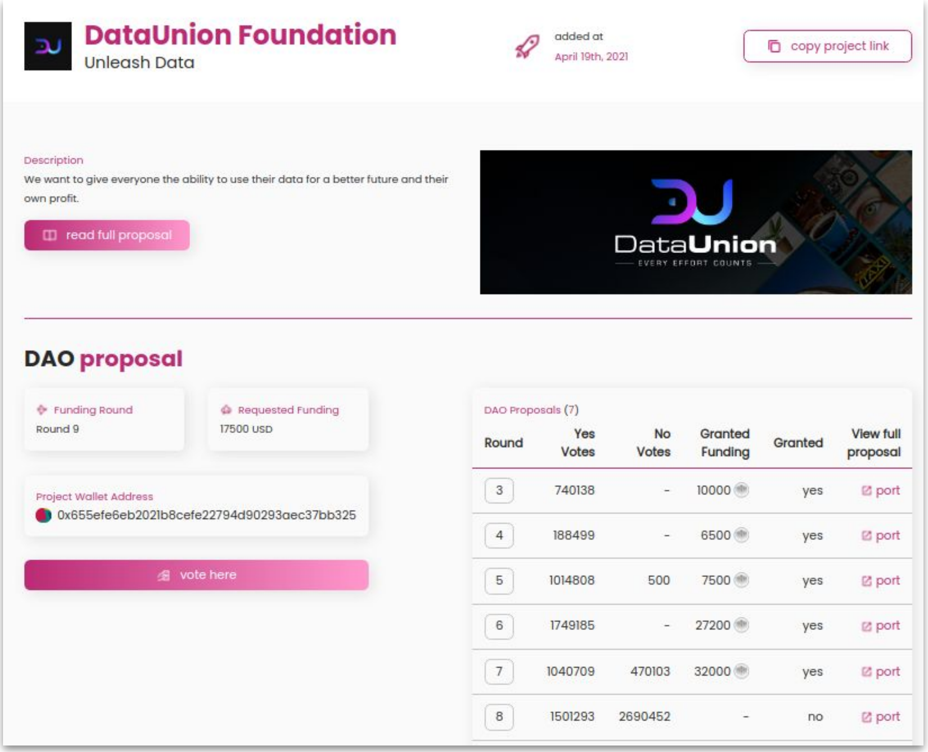
70.067 OCEAN POOL
≈ €40.56

BUY FOR 1 YEAR

For using this dataset, you will buy 1 TREPEL-36 and immediately spend it back to the publisher and pool.

🟥 **No account connected**
Please connect your Web3 wallet.

DataUnion: AI Image Labeling Union



DataUnion Foundation

Unleash Data

added at
April 19th, 2021

copy project link

Description

We want to give everyone the ability to use their data for a better future and their own profit.

read full proposal

DAO proposal

Funding Round
Round 9

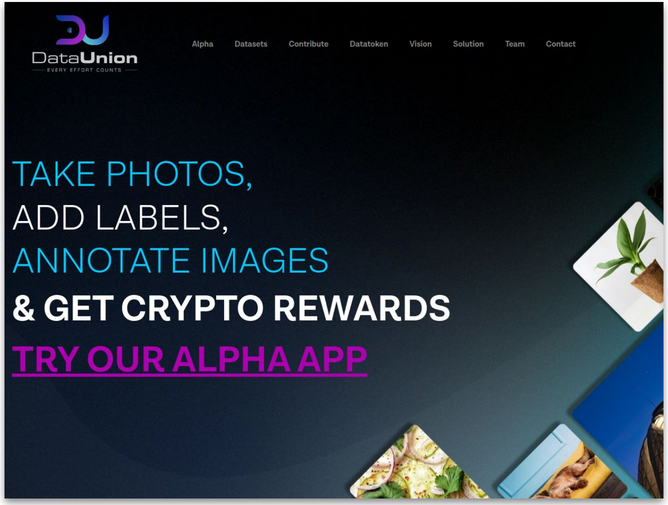
Requested Funding
17500 USD

Project Wallet Address
0x655efe6eb2021b8cefe22794d90293aec37bb325

vote here

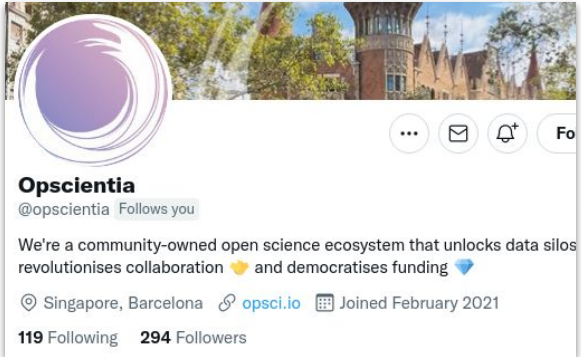
DAO Proposals (7)

Round	Yes Votes	No Votes	Granted Funding	Granted	View full proposal
3	740138	-	10000	yes	port
4	188499	-	6500	yes	port
5	1014808	500	7500	yes	port
6	1749185	-	27200	yes	port
7	1040709	470103	32000	yes	port
8	1501293	2690452	-	no	port



TAKE PHOTOS,
ADD LABELS,
ANNOTATE IMAGES
& GET CRYPTO REWARDS
TRY OUR ALPHA APP

Opscientia: Data DAO around Open Science

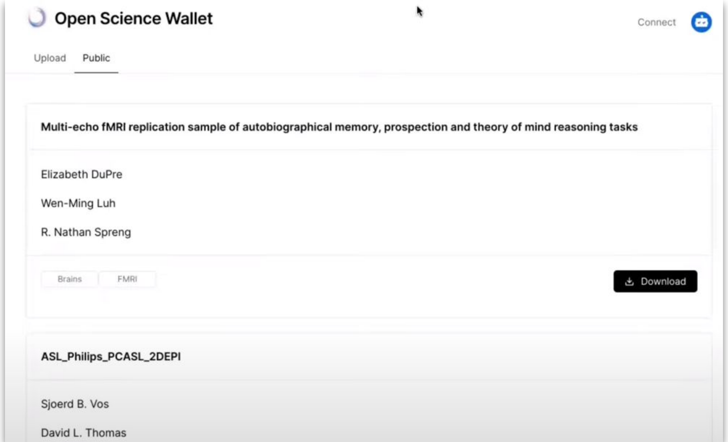


Opscientia
@opscientia Follows you

We're a community-owned open science ecosystem that unlocks data silos revolutionises collaboration 🤝 and democratises funding 💎

Singapore, Barcelona opsci.io Joined February 2021

119 Following **294** Followers



Multi-echo fMRI replication sample of autobiographical memory, prospection and theory of mind reasoning tasks

Elizabeth DuPre

Wen-Ming Luh

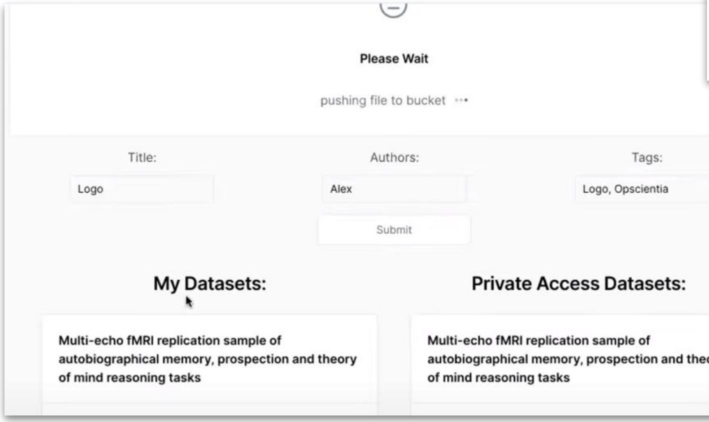
R. Nathan Spreng

Brains FMRI

ASL_Philips_PCASL_2DEPI

Sjoerd B. Vos

David L. Thomas



Please Wait

pushing file to bucket •••

Title: Logo

Authors: Alex

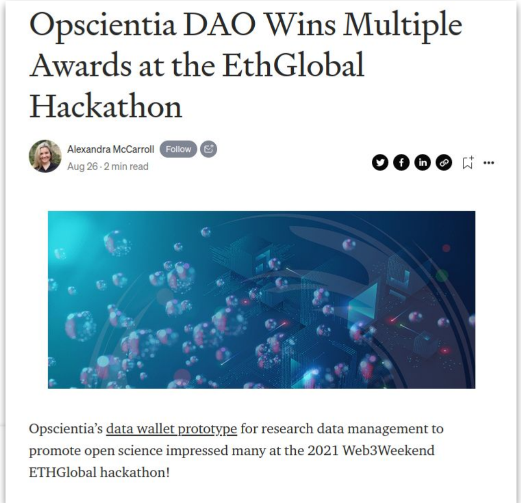
Tags: Logo, Opscientia

My Datasets:

Multi-echo fMRI replication sample of autobiographical memory, prospection and theory of mind reasoning tasks

Private Access Datasets:

Multi-echo fMRI replication sample of autobiographical memory, prospection and theory of mind reasoning tasks



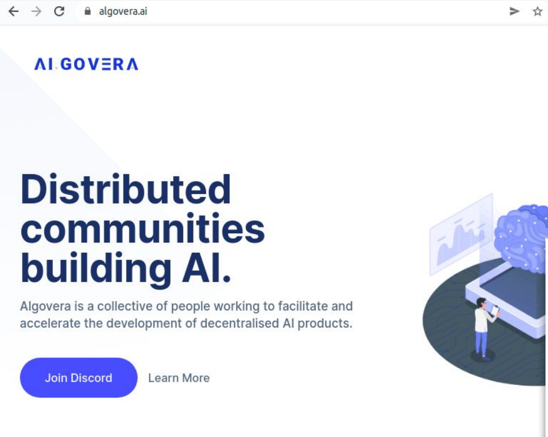
Opscientia DAO Wins Multiple Awards at the EthGlobal Hackathon

Alexandra McCarroll
Aug 26 · 2 min read

Opscientia's data wallet prototype for research data management to promote open science impressed many at the 2021 Web3Weekend ETHGlobal hackathon!

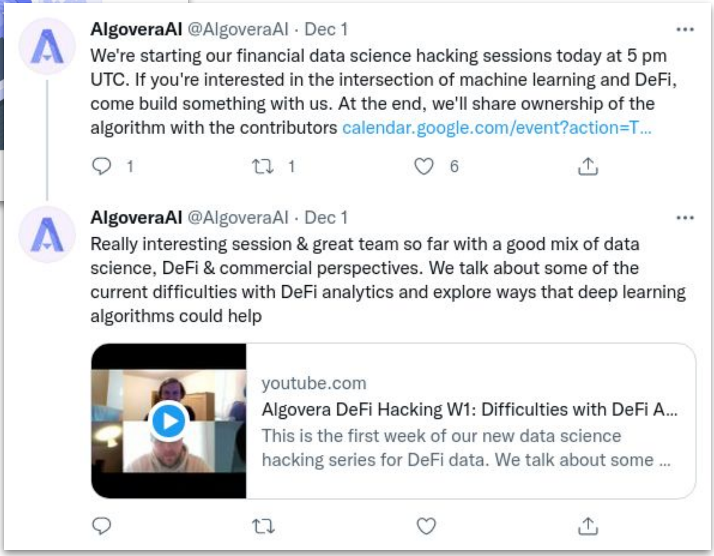
Algovera: AI community * Ocean

algovera.ai



Distributed communities building AI.

Algovera is a collective of people working to facilitate and accelerate the development of decentralised AI products.



AlgoveraAI @AlgoveraAI · Dec 1

We're starting our financial data science hacking sessions today at 5 pm UTC. If you're interested in the intersection of machine learning and DeFi, come build something with us. At the end, we'll share ownership of the algorithm with the contributors calendar.google.com/event?action=T...

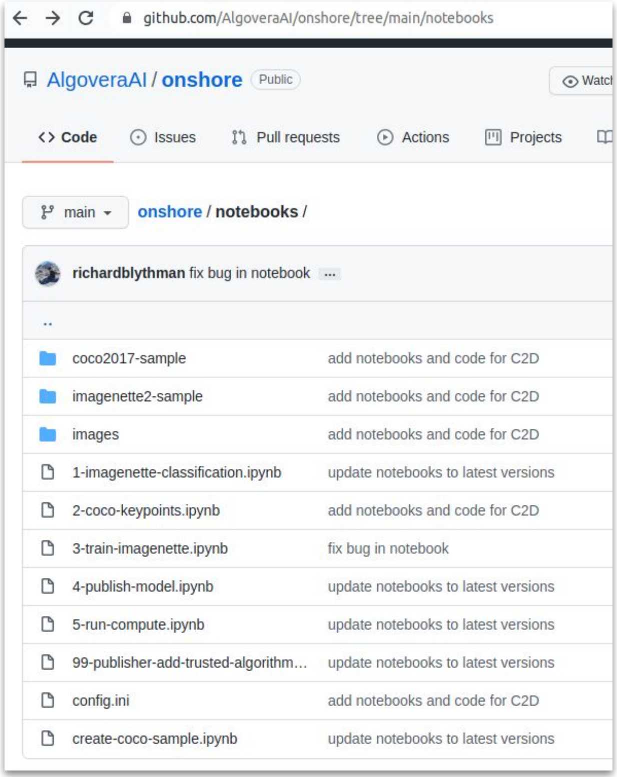
AlgoveraAI @AlgoveraAI · Dec 1

Really interesting session & great team so far with a good mix of data science, DeFi & commercial perspectives. We talk about some of the current difficulties with DeFi analytics and explore ways that deep learning algorithms could help

youtube.com

Algovera DeFi Hacking W1: Difficulties with DeFi A...

This is the first week of our new data science hacking series for DeFi data. We talk about some ...


github.com/AlgoveraAI/onshore/tree/main/notebooks
AlgoveraAI / onshore Public
Watch
Code
Issues
Pull requests
Actions
Projects
main

onshore / notebooks /

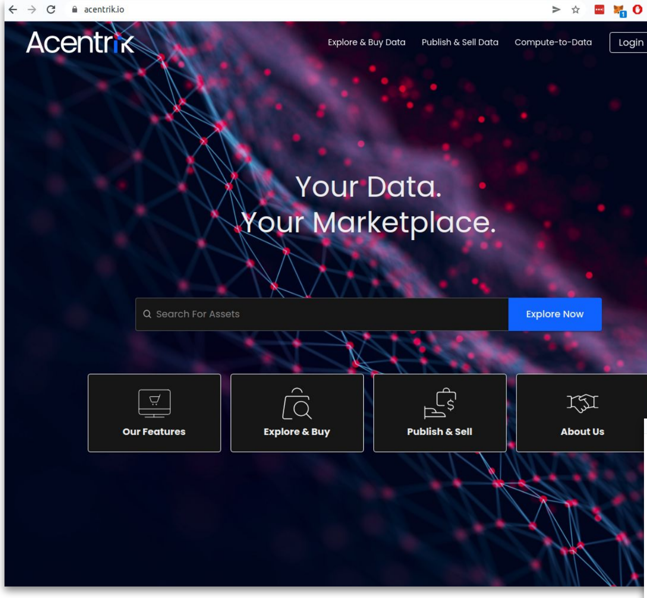
richardblythman fix bug in notebook ...

..

coco2017-sample	add notebooks and code for C2D
imagenette2-sample	add notebooks and code for C2D
images	add notebooks and code for C2D
1-imagenette-classification.ipynb	update notebooks to latest versions
2-coco-keypoints.ipynb	add notebooks and code for C2D
3-train-imagenette.ipynb	fix bug in notebook
4-publish-model.ipynb	update notebooks to latest versions
5-run-compute.ipynb	update notebooks to latest versions
99-publisher-add-trusted-algorithm...	update notebooks to latest versions
config.ini	add notebooks and code for C2D
create-coco-sample.ipynb	update notebooks to latest versions

For an *economy*, 1000s of data marketplaces

Here are some examples



acentrik.io

Acentrik

Explore & Buy Data

Publish & Sell Data

Compute-to-Data

Login

Sign Up

Your Data.
Your Marketplace.

Search For Assets

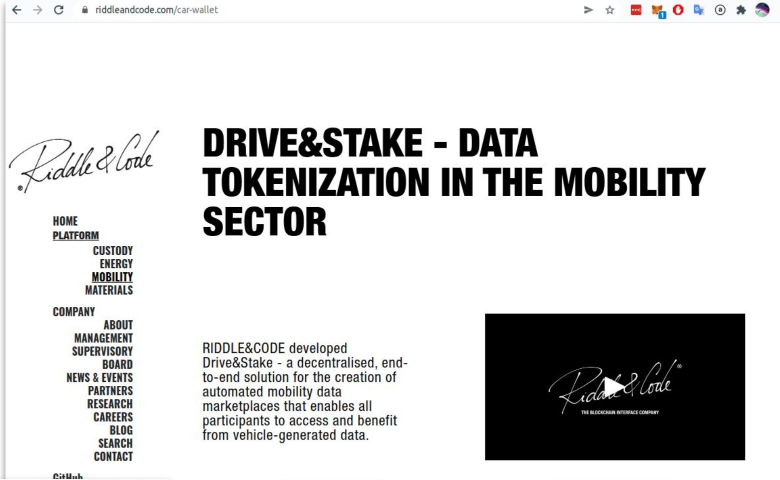
Explore Now

Our Features

Explore & Buy

Publish & Sell

About Us



riddleandcode.com/car-wallet

Riddle & Code

HOME
PLATFORM
CUSTODY
ENERGY
MOBILITY
MATERIALS
COMPANY
ABOUT
MANAGEMENT
SUPERVISORY
BOARD
NEWS & EVENTS
PARTNERS
RESEARCH
CAREERS
BLOG
SEARCH
CONTACT
GitHub

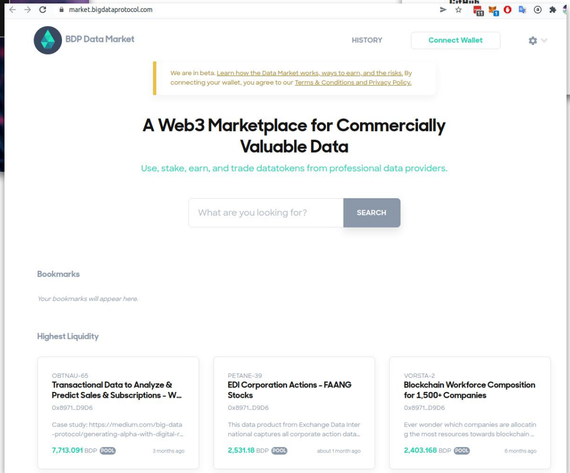
DRIVE&STAKE - DATA TOKENIZATION IN THE MOBILITY SECTOR

RIDDLE&CODE developed Drive&Stake - a decentralised, end-to-end solution for the creation of automated mobility data marketplaces that enables all participants to access and benefit from vehicle-generated data.

Riddle & Code
THE BLOCKCHAIN INTERFACE COMPANY

LET'S TALK

PRESS RELEASE



market.bigdataprotocol.com

BDP Data Market

HISTORY

Connect Wallet

We are in beta. Learn how the Data Market works, ways to earn, and the risks. By connecting your wallet, you agree to our Terms & Conditions and Privacy Policy.

A Web3 Marketplace for Commercially Valuable Data

Use, stake, earn, and trade datatokens from professional data providers.

What are you looking for?

SEARCH

Bookmarks

Your bookmarks will appear here.

Highest Liquidity

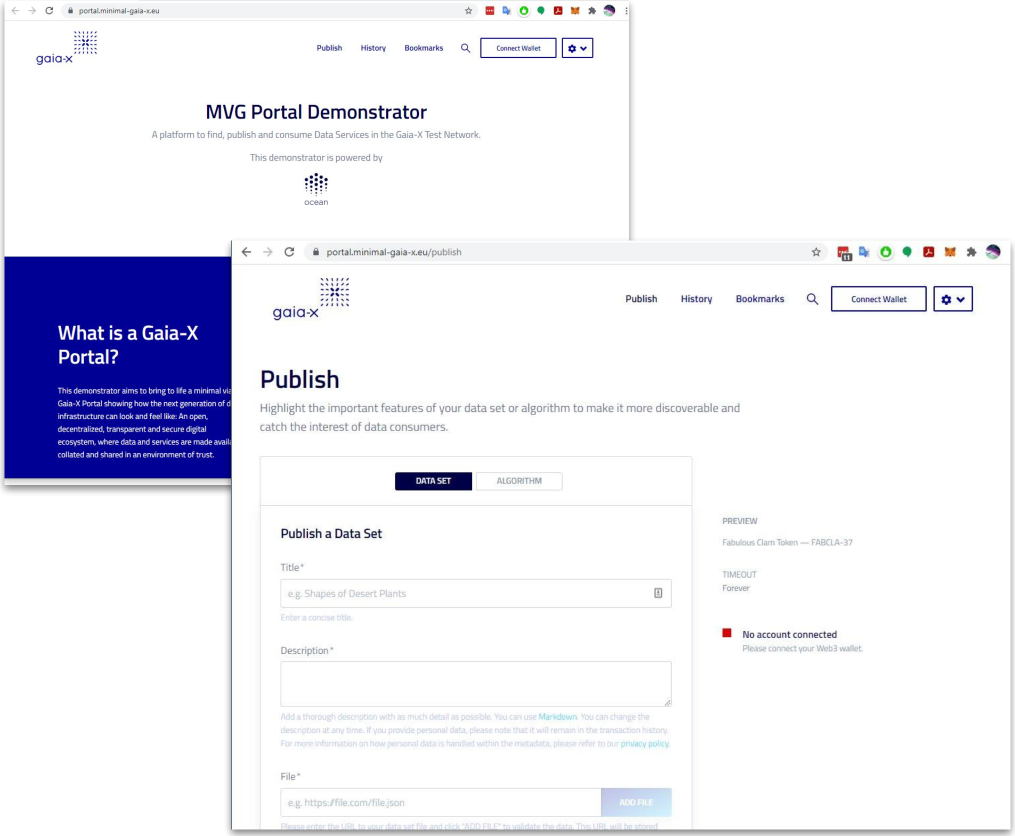
OBTNAU-65
Transactional Data to Analyze & Predict Sales & Subscriptions - W...
0x8971...D9D6
Case study: https://medium.com/big-data-protocol/generating-alpha-with-digital-r...
7,713.091 BDP POOL 3 months ago

PETANE-39
EDI Corporation Actions - FAANG Stocks
0x8971...D9D6
This data product from Exchange Data International captures all corporate action data...
2,531.18 BDP POOL about 1 month ago

VORSTA-2
Blockchain Workforce Composition for 1,500+ Companies
0x8971...D9D6
Ever wonder which companies are allocating the most resources towards blockchain ...
2,403.168 BDP POOL 6 months ago

DeltaDAO: Gaia-X Demonstrator

https://portal.minimal-gaia-x.eu



portal.minimal-gaia-x.eu

MVG Portal Demonstrator

A platform to find, publish and consume Data Services in the Gaia-X Test Network.

This demonstrator is powered by

What is a Gaia-X Portal?

This demonstrator aims to bring to life a minimal via
Gaia-X Portal showing how the next generation of d
infrastructure can look and feel like: An open,
decentralized, transparent and secure digital
ecosystem, where data and services are made availa
collated and shared in an environment of trust.

portal.minimal-gaia-x.eu/publish

Publish

Highlight the important features of your data set or algorithm to make it more discoverable and catch the interest of data consumers.

Publish a Data Set

Title*

e.g. Shapes of Desert Plants

Enter a concise title.

Description*

Add a thorough description with as much detail as possible. You can use Markdown. You can change the description at any time. If you provide personal data, please note that it will remain in the transaction history. For more information on how personal data is handled within the metadata, please refer to our privacy policy.

File*

e.g. https://file.com/file.json

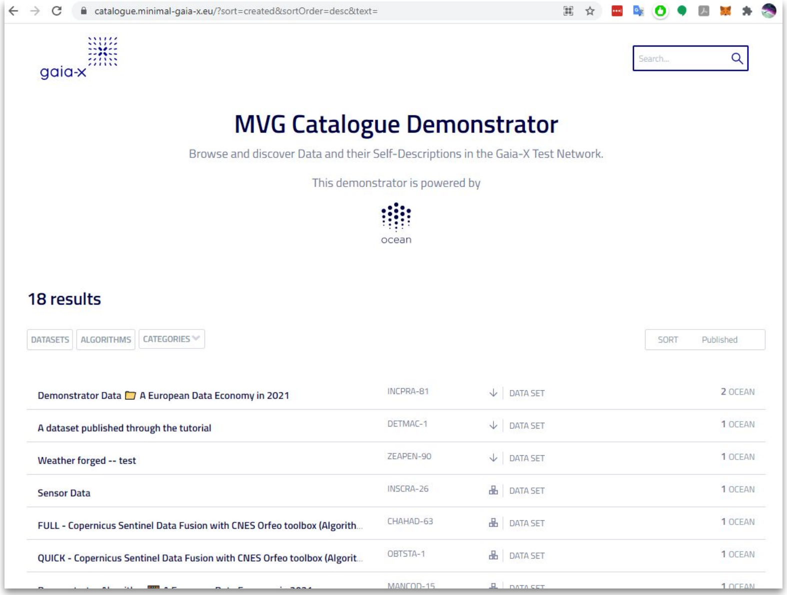
Please enter the URL to your data set file and click "ADD FILE" to validate the data. This URL will be stored

PREVIEW

Fabulous Clam Token — FABCLA-37

TIMEOUT
Forever

🟥 No account connected
Please connect your Web3 wallet.



catalogue.minimal-gaia-x.eu/?sort=created&sortOrder=desc&text=

MVG Catalogue Demonstrator

Browse and discover Data and their Self-Descriptions in the Gaia-X Test Network.

This demonstrator is powered by

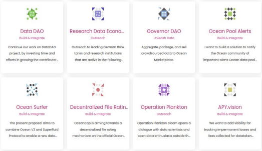
18 results

Demonstrator Data 📁 A European Data Economy in 2021	INCPRA-81	DATA SET	2 OCEAN
A dataset published through the tutorial	DETMAC-1	DATA SET	1 OCEAN
Weather forged -- test	ZEAPEN-90	DATA SET	1 OCEAN
Sensor Data	INSCRA-26	DATA SET	1 OCEAN
FULL - Copernicus Sentinel Data Fusion with CNES Orfeo toolbox (Algorith...	CHAHAD-63	DATA SET	1 OCEAN
QUICK - Copernicus Sentinel Data Fusion with CNES Orfeo toolbox (Algorit...	OBTSTA-1	DATA SET	1 OCEAN
Demonstrator Algorithm A European Data Economy in 2021	MANCOD-15	DATA SET	1 OCEAN

ocean

OceanDAO

oceanpearl.io/projects



Data DAO
Build & Integrate
Continue our work on DataDAO project, by investing time and efforts in growing the contributor...

Research Data Econo...
Outreach
Outreach to leading German think tanks and research institutions that are active in the following...

Governor DAO
Unleash Data
Aggregate, package, and sell crowdsourced data to Ocean Marketplace.

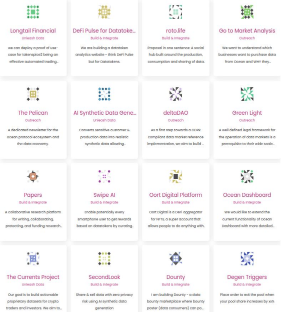
Ocean Pool Alerts
Build & Integrate
I want to build a solution to notify the Ocean community of important alerts Ocean data pool...

Ocean Surfer
Build & Integrate
The present proposal aims to combine Ocean V3 and Superfluid Protocol to enable a new data...

Decentralized File Ratin...
Build & Integrate
Oceancap is aiming towards a decentralized file rating mechanism on the official Ocean...

Operation Plankton
Outreach
Operation Plankton Bloom opens a dialogue with data scientists and open data enthusiasts outside th...

APY.vision
Build & Integrate
We want to add visbility for tracking impermanent losses and fees collected for datatoken...



Longtail Financial
Unleash Data
we can deploy a proof of use-case for tokenspice2 being an effective automated trading...

DeFi Pulse for Datatoke...
Build & Integrate
We are building a datatoken analytics website - think DeFi Pulse but for Datatokens.

roto.life
Build & Integrate
Proposal in one sentence: A social hub built around the production, consumption and sharing of data.

Go to Market Analysis
Outreach
We want to understand which businesses want to purchase data from Ocean and WHY they...

The Pelican
Outreach
A dedicated newsletter for the ocean protocol ecosystem and the data economy.

AI Synthetic Data Gene...
Unleash Data
Converts sensitive customer & production data into realistic synthetic data allowing...

deltaDAO
Outreach
As a first step towards a GDPR compliant data market reference implementation, we aim to build ...

Green Light
Outreach
A well defined legal framework for the operation of data markets is a prerequisite to their wide scale...

Papers
Build & Integrate
A collaborative research platform for writing, collaborating, protecting, and funding research...

Swipe AI
Build & Integrate
Enable potentially every smartphone user to get rewards based on datatokens by curating...

Oort Digital Platform
Build & Integrate
Oort Digital is a DeFi aggregator for NFTs, a super account that allows people to do anything with...

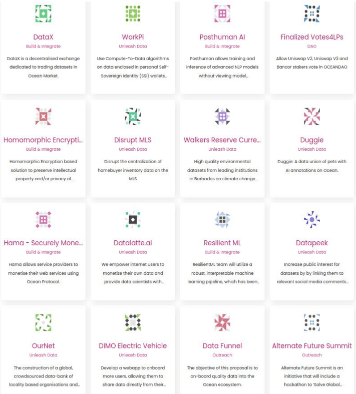
Ocean Dashboard
Build & Integrate
We would like to extend the current functionality of Ocean Dashboard with more detailed...

The Currents Project
Unleash Data
Our goal is to build actionable proprietary datasets for crypto traders and investors. We aim to...

SecondLook
Build & Integrate
Share & sell data with zero privacy risk using AI synthetic data generation

Dounty
Build & Integrate
I am building Dounty - a data bounty marketplace where bounty poster (data consumers) can po...

Degen Triggers
Build & Integrate
Place order to exit the pool when your pool share increases by xx%



DataX
Build & Integrate
DataX is a decentralised exchange dedicated to trading datasets in Ocean Market.

WorkPi
Unleash Data
Use Compute-To-Data algorithms on data enclosed in personal Self-Sovereign Identity (SSI) wallets...

Posthuman AI
Build & Integrate
Posthuman allows training and inference of advanced NLP models without viewing model...

Finalized Votes4LPs
DAO
Allow Uniswap V2, Uniswap V3 and Bancor stakers vote in OCEANDAO

Homomorphic Encrypti...
Build & Integrate
Homomorphic Encryption based solution to preserve intellectual property and/or privacy of...

Disrupt MLS
Unleash Data
Disrupt the centralization of homebuyer inventory data on the MLS

Walkers Reserve Curre...
Unleash Data
High quality environmental datasets from leading institutions in Barbados on climate change...

Duggie
Unleash Data
Duggie: A data union of pets with AI annotations on Ocean.

Hama - Securely Mone...
Build & Integrate
Hama allows service providers to monetise their web services using Ocean Protocol.

Datalatte.ai
Unleash Data
We empower internet users to monetize their own data and provide data scientists with...

Resilient ML
Build & Integrate
ResilientML team will utilize a robust, interpretable machine learning pipeline, which has been...

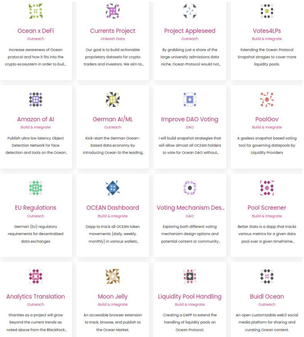
Datapeek
Unleash Data
Increase public interest for datasets by by linking them to relevant social media comments...

OurNet
Unleash Data
The construction of a global, crowdsourced data-bank of locality based organisations and...

DIMO Electric Vehicle
Unleash Data
Develop a webapp to onboard more users, allowing them to share data directly from their...

Data Funnel
Outreach
The objective of this proposal is to on-board quality data into the Ocean ecosystem.

Alternate Future Summit
Outreach
Alternate Future Summit is an initiative that will include a hackathon to 'Solve Global...



Ocean x DeFi
Outreach
Increase awareness of Ocean protocol and how it fits into the crypto ecosystem in order to buil...

Currents Project
Unleash Data
Our goal is to build actionable proprietary datasets for crypto traders and investors. We aim to...

Project Appleseed
Outreach
By grabbing just a share of the large university admissions data niche, Ocean Protocol would not...

Votes4LPs
Build & Integrate
Extending the Ocean Protocol Snapshot stragies to cover more liquidity pools.

Amazon of AI
Build & Integrate
Publish ultra low latency Object Detection Network for face detection and tools on the Ocean...

German AI/ML
Outreach
Kick-start the German Ocean-based data economy by introducing Ocean to the leading...

Improve DAO Voting
DAO
I will build snapshot strategies that will allow almost all OCEAN holders to vote for Ocean DAO without...

PoolGov
Build & Integrate
A gasless snapshot based voting tool for governing datapools by Liquidity Providers

EU Regulations
Outreach
German (EU) regulatory requirements for decentralized data exchanges

OCEAN Dashboard
Build & Integrate
Dapp to track all OCEAN token movements (daily, weekly, monthly) in various wallets...

Voting Mechanism Des...
DAO
Exploring both different voting mechanism design options and potential content or community...

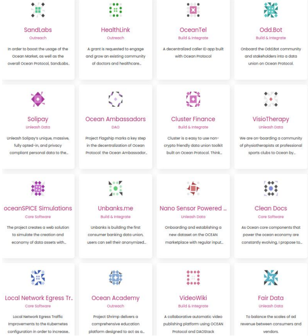
Pool Screener
Build & Integrate
Better Stats is a dapp that tracks various metrics for a given data pool over a given timeframe...

Analytics Translation
Outreach
Shanties as a project will grow beyond the current trends as noted above from the BlackRock...

Moon Jelly
Build & Integrate
An accessible browser extension to track, browse, and publish to the Ocean Market.

Liquidity Pool Handling
Build & Integrate
Creating a DAPP to extend the handling of liquidity pools on Ocean Protocol.

Buidl Ocean
Outreach
An open customizable web3 social media platform for sharing and curating Ocean content.



SandLabs
Outreach
In order to boost the usage of the Ocean Market, as well as the overall Ocean Protocol, SandLabs...

HealthLink
Outreach
A grant is requested to engage and grow an existing community of doctors and healthcare...

OceanTel
Build & Integrate
A decentralized caller ID app built with Ocean Protocol

Odd.Bot
Build & Integrate
Onboard the Odd.Bot community and stakeholders into a data union on Ocean Protocol.

Solipay
Unleash Data
Unleash Solipay's unique, massive, fully opted-in, and privacy compliant personal data to the...

Ocean Ambassadors
DAO
Project Flagship marks a key step in the decentralization of Ocean Protocol: the Ocean Ambassador...

Cluster Finance
Build & Integrate
Cluster is a easy to use non-crypto friendly data union toolkit built on Ocean Protocol. Think...

VisioTherapy
Unleash Data
We are on-boarding a community of physiotherapists at professional sports clubs to Ocean by...

oceanSPICE Simulations
Core Software
The project creates a web solution to simulate the creation and economy of data assets with...

Unbanks.me
Build & Integrate
Unbanks is building the first consumer banking data union, users can sell their anonymized...

Nano Sensor Powered ...
Unleash Data
Onboarding and establishing a new dataset on the OCEAN marketplace with regular input...

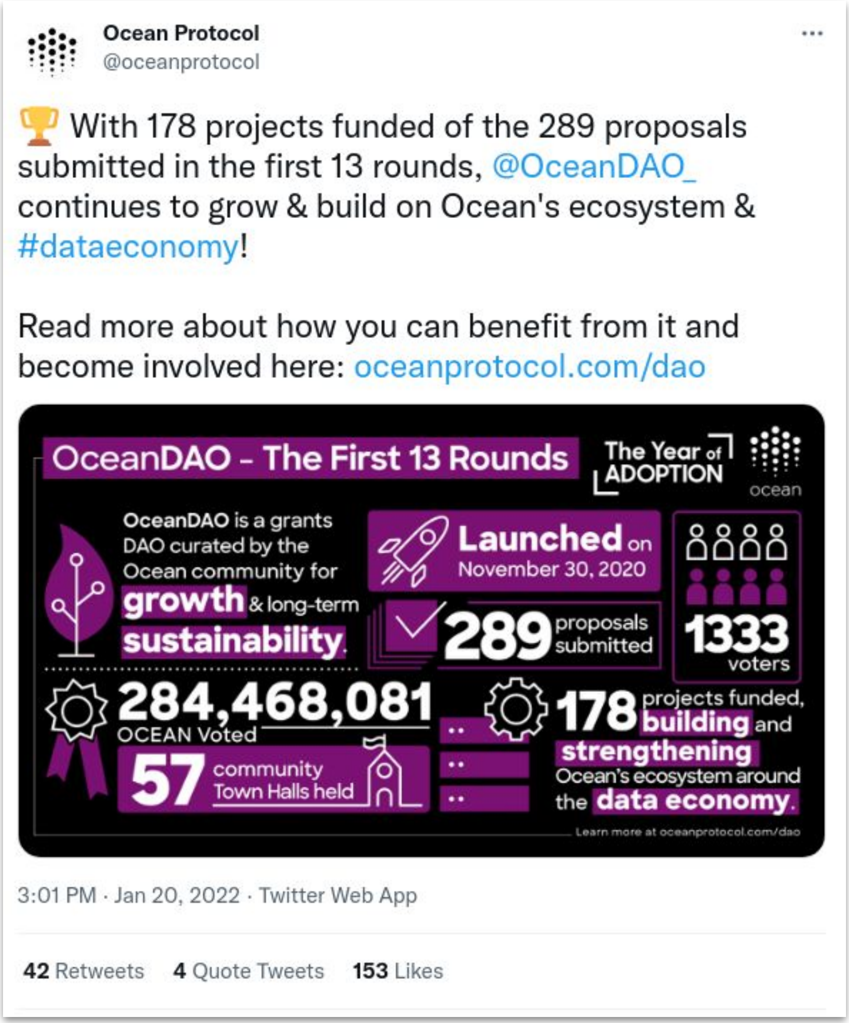
Clean Docs
Core Software
As Ocean core components that power the ocean economy are constantly evolving, I propose to...

Local Network Egress Tr...
Core Software
Local Network Egress Traffic Improvements to the Kubernetes configuration in order to increase...

Ocean Academy
Outreach
Project Shrimp delivers a comprehensive education platform designed to act as a...

VideoWiki
Build & Integrate
A collaborative automatic video publishing platform using OCEAN Protocol and DAOStack

Fair Data
Unleash Data
To balance the scales of ad revenue between consumers and vendors.



Ocean Protocol
@oceanprotocol

🏆 With 178 projects funded of the 289 proposals submitted in the first 13 rounds, @OceanDAO_ continues to grow & build on Ocean's ecosystem & #dataeconomy!

Read more about how you can benefit from it and become involved here: oceanprotocol.com/dao

OceanDAO – The First 13 Rounds

The Year of ADOPTION

ocean

OceanDAO is a grants DAO curated by the Ocean community for **growth** & long-term **sustainability**.

Launched on November 30, 2020

289 proposals submitted

1333 voters

284,468,081 OCEAN Voted

57 community Town Halls held

178 projects funded, **building** and **strengthening** Ocean's ecosystem around the **data economy**.

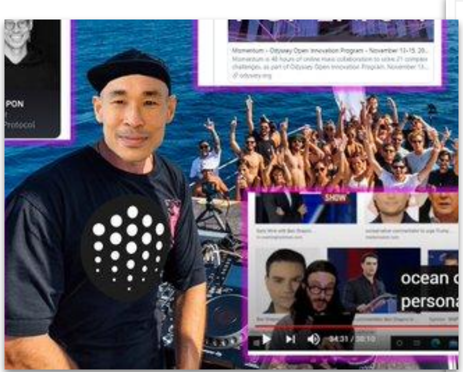
Learn more at oceanprotocol.com/dao

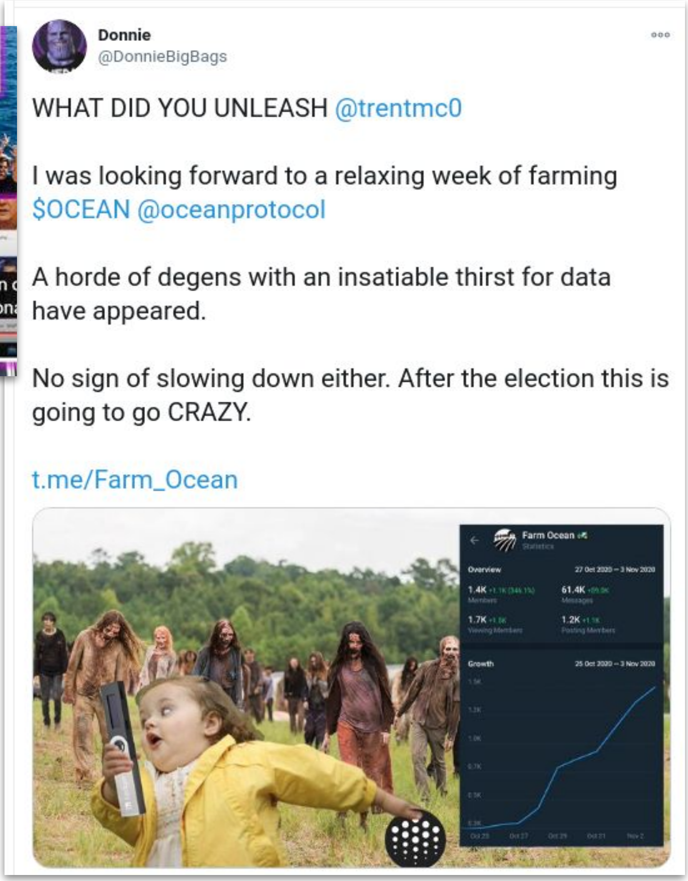
3:01 PM · Jan 20, 2022 · Twitter Web App

42 Retweets **4** Quote Tweets **153** Likes

Community Memes / Art

Ocean data farmers can earn OCEAN for providing liquidity to OCEAN-datatoken pools; the amount earned will be multiplied by usage of the datasets they stake on, and more.





Donnie
@DonnieBigBags

WHAT DID YOU UNLEASH @trentmc0

I was looking forward to a relaxing week of farming $OCEAN @oceanprotocol

A horde of degens with an insatiable thirst for data have appeared.

No sign of slowing down either. After the election this is going to go CRAZY.

t.me/Farm_Ocean

Some Ocean Collaborations

ITU

gaia-x

XPRIZE

Reflexer

WORLD ECONOMIC FORUM

GITCOIN

BINANCE

DAIMLER

FORESIGHT INSTITUTE

energy web FOUNDATION

GridSingularity

Chainlink

Balancer

NEWORDER

RADICALxCHANGE

MOBI

MESSARI

Filecoin

Data Services in Ocean

ocean

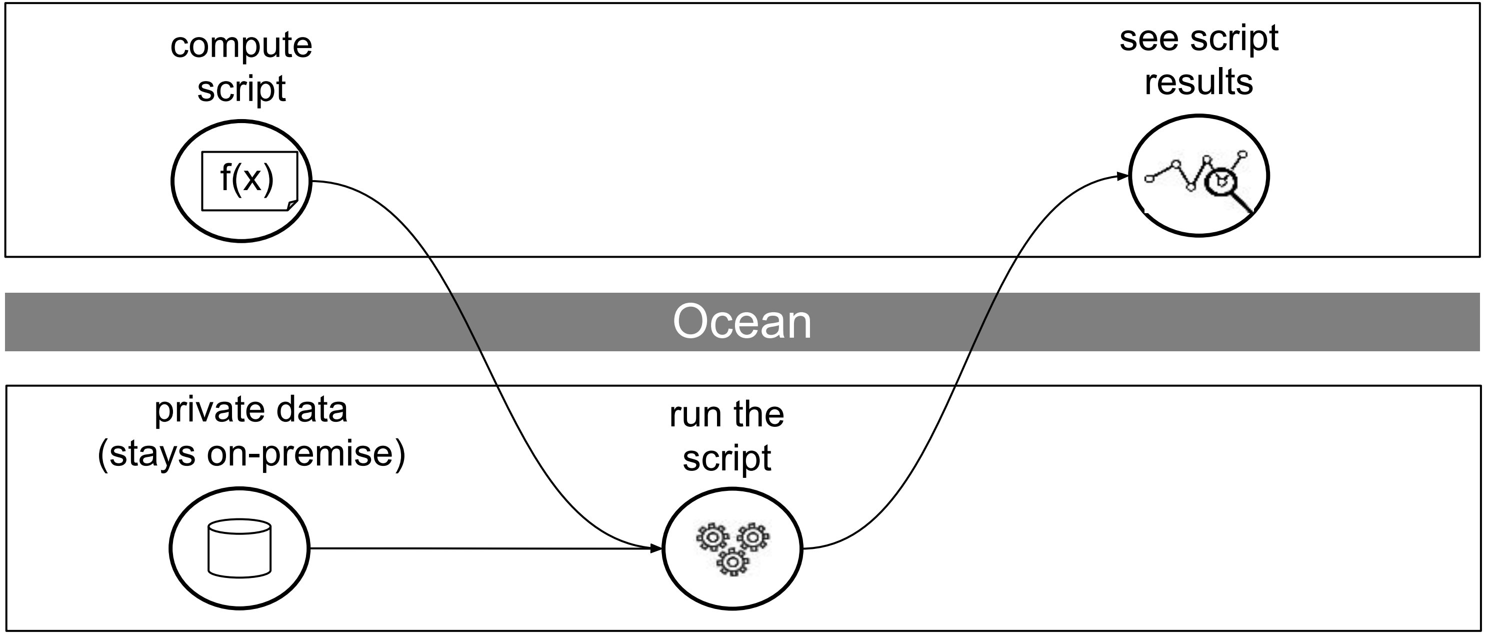
Ocean Data Service #1: Static Uri

- Publisher encrypts the uri
- Uri gets decrypted upon ‘consume’
- Available in backend and frontend (Ocean Market)

- Works for...
  - static files
  - dynamically-updated files
  - for Web2 REST APIs
  - and more. Super-flexible.

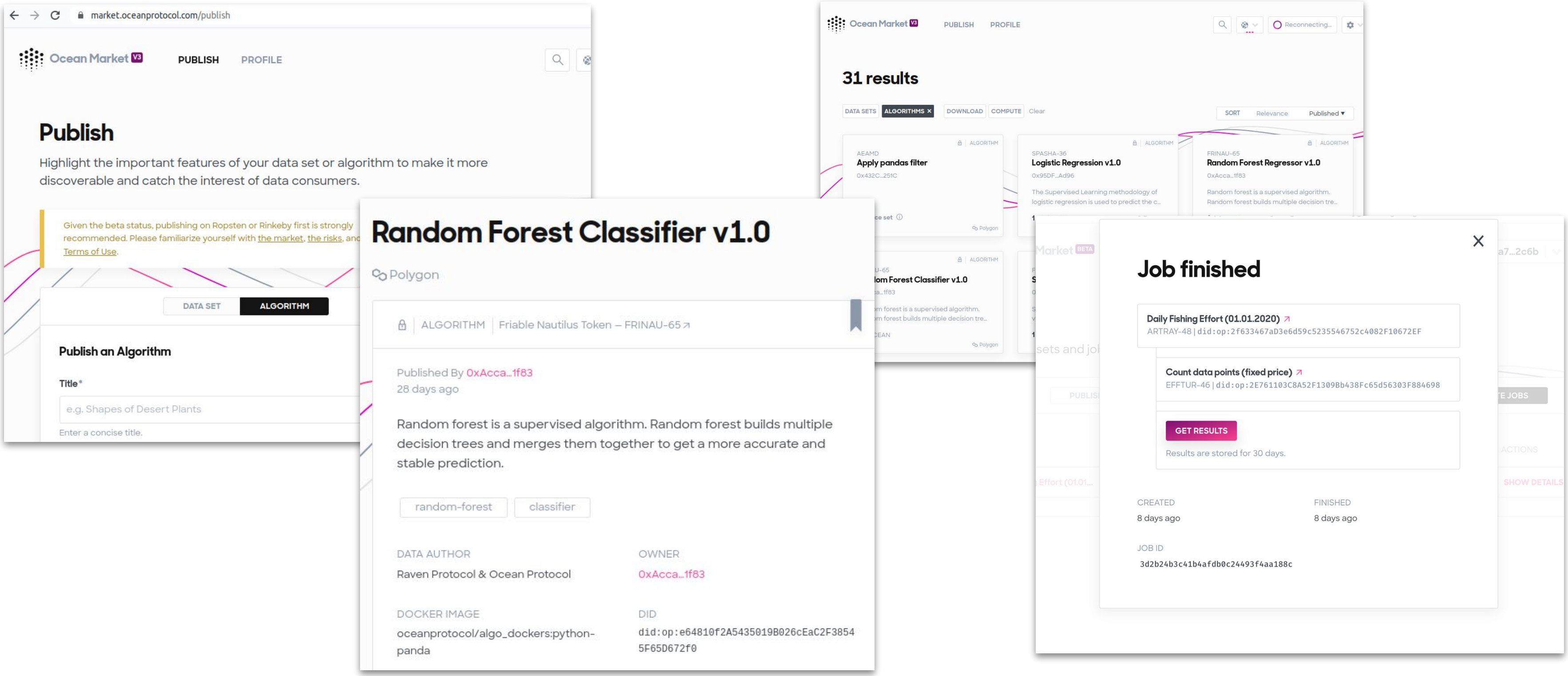
Ocean Data Service #2: Compute-to-Data (C2D)

Buy & sell private data, while preserving privacy



Compute-to-Data (C2D) In Ocean Market

blog.oceanprotocol.com/compute-to-data-is-now-available-in-ocean-market-58868be52ef7



market.oceanprotocol.com/publish

Publish

Highlight the important features of your data set or algorithm to make it more discoverable and catch the interest of data consumers.

Given the beta status, publishing on Ropsten or Rinkeby first is strongly recommended. Please familiarize yourself with the market, the risks, and Terms of Use.

Publish an Algorithm

Title*

e.g. Shapes of Desert Plants

Enter a concise title.

Random Forest Classifier v1.0

Polygon

ALGORITHM | Friable Nautilus Token – FRINAU-65

Published By 0xAcca…1f83
28 days ago

Random forest is a supervised algorithm. Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

DATA AUTHOR	OWNER
Raven Protocol & Ocean Protocol	0xAcca…1f83
DOCKER IMAGE	**DID**
oceanprotocol/algo_dockers:python-panda	did:op:e64810f2A5435019B026cEaC2F38545F65D672f0

31 results

ALGORITHM
AEAMD
Apply pandas filter
0x432C…251C
Polygon

ALGORITHM
SPASHA-36
Logistic Regression v1.0
0x95DF…Ad96
The Supervised Learning methodology of logistic regression is used to predict the c…

ALGORITHM
FRINAU-65
Random Forest Regressor v1.0
0xAcca…1f83
Random forest is a supervised algorithm. Random forest builds multiple decision tre…

ALGORITHM
U-65
lom Forest Classifier v1.0
a…1f83
m forest is a supervised algorithm.
m forest builds multiple decision tre…
Polygon

Job finished

Daily Fishing Effort (01.01.2020)
ARTRAY-48 | did:op:2f633467aD3e6d59c5235546752c4082F10672EF

Count data points (fixed price)
EFFTUR-46 | did:op:2E761103C8A52F1309Bb438Fc65d56303F884698

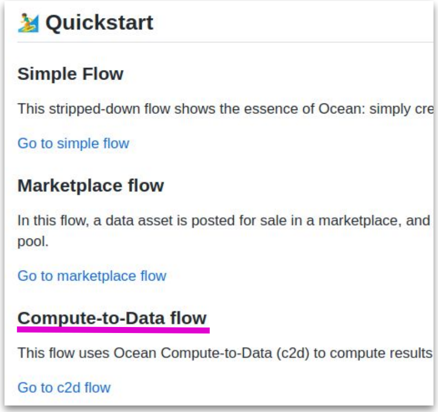
Results are stored for 30 days.

CREATED	FINISHED
8 days ago	8 days ago

JOB ID
3d2b24b3c41b4afdb0c24493f4aa188c

C2D Quickstart via Ocean.py: Overview

github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



🏄 Quickstart

Simple Flow

This stripped-down flow shows the essence of Ocean: simply cre

Go to simple flow

Marketplace flow

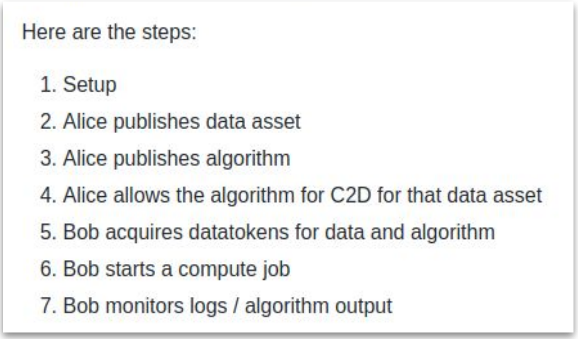
In this flow, a data asset is posted for sale in a marketplace, and pool.

Go to marketplace flow

Compute-to-Data flow

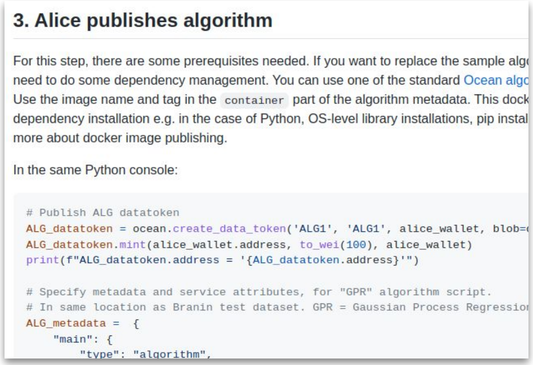
This flow uses Ocean Compute-to-Data (c2d) to compute results

Go to c2d flow



Here are the steps:

1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

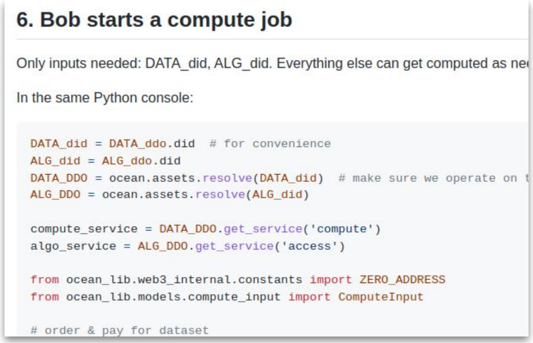


3. Alice publishes algorithm

For this step, there are some prerequisites needed. If you want to replace the sample alg
need to do some dependency management. You can use one of the standard Ocean alg
Use the image name and tag in the `container` part of the algorithm metadata. This dock
dependency installation e.g. in the case of Python, OS-level library installations, pip instal
more about docker image publishing.

In the same Python console:

```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob=
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regressio
ALG_metadata =  {
    "main": {
        "type": "algorithm",
```

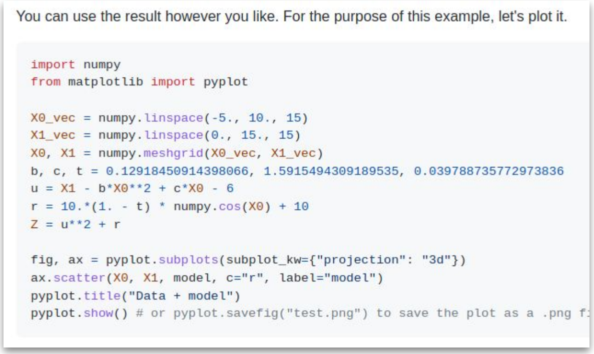


6. Bob starts a compute job

Only inputs needed: DATA_did, ALG_did. Everything else can get computed as ne

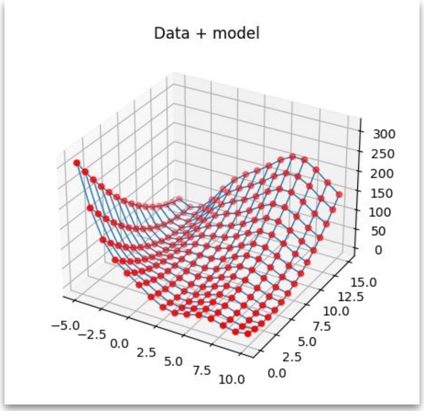
In the same Python console:

```
DATA_did = DATA_ddo.did  # for convenience
ALG_did = ALG_ddo.did
DATA_DDO = ocean.assets.resolve(DATA_did)  # make sure we operate on t
ALG_DDO = ocean.assets.resolve(ALG_did)

compute_service = DATA_DDO.get_service('compute')
algo_service = ALG_DDO.get_service('access')

from ocean_lib.web3_internal.constants import ZERO_ADDRESS
from ocean_lib.models.compute_input import ComputeInput

# order & pay for dataset
```



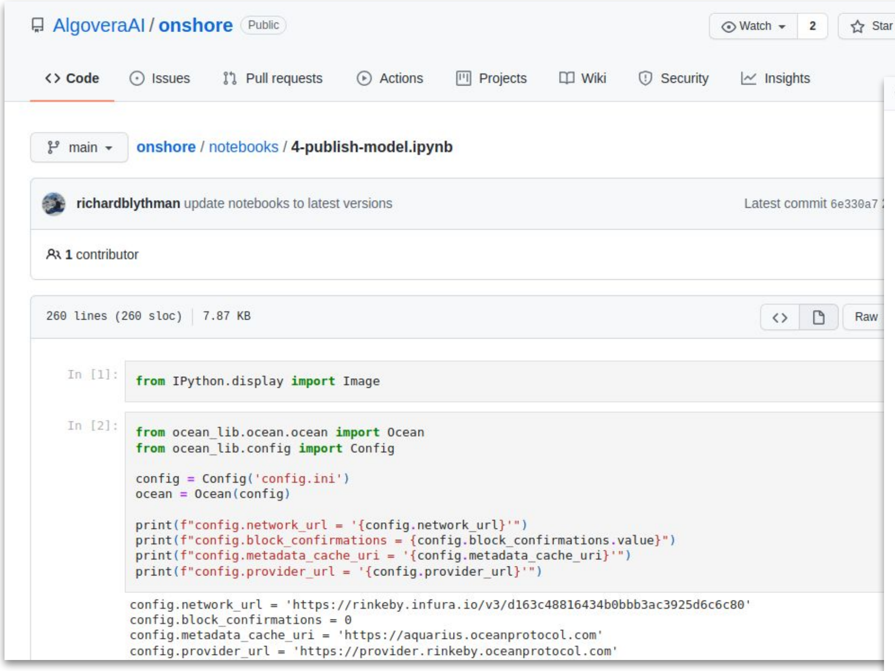
You can use the result however you like. For the purpose of this example, let's plot it.

```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.039788735772973836
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png f
```



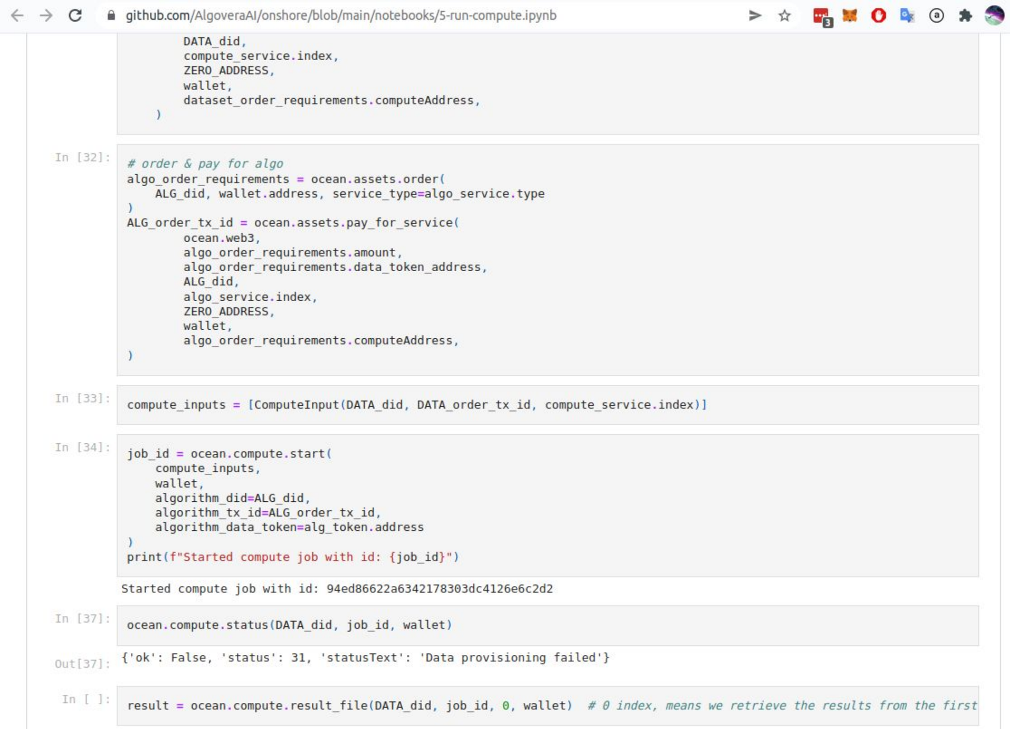
C2D * Jupyter: Publish Model, Consume Model

https://github.com/AlgoveraAI/onshore/blob/main/notebooks/4-publish-model.ipynb Thanks, Algovera!


AlgoveraAI / onshore Public
Watch 2
Star 1
Fork 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
main
onshore / notebooks / 4-publish-model.ipynb
richardblythman update notebooks to latest versions
Latest commit 6e330a7
1 contributor
260 lines (260 sloc) | 7.87 KB
Raw

```
In [1]:  from IPython.display import Image
```

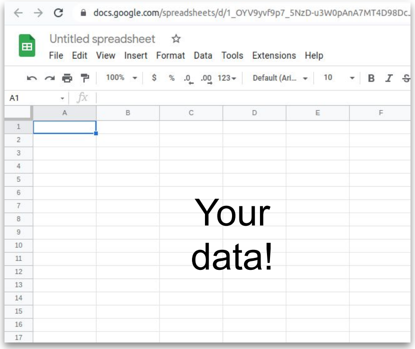
```
In [2]:  from ocean_lib.ocean.ocean import Ocean
         from ocean_lib.config import Config

         config = Config('config.ini')
         ocean = Ocean(config)

         print(f"config.network_url = '{config.network_url}'")
         print(f"config.block_confirmations = {config.block_confirmations.value}")
         print(f"config.metadata_cache_uri = '{config.metadata_cache_uri}'")
         print(f"config.provider_url = '{config.provider_url}'")

         config.network_url = 'https://rinkeby.infura.io/v3/d163c48816434b0bbb3ac3925d6c6c80'
         config.block_confirmations = 0
         config.metadata_cache_uri = 'https://aquarius.oceanprotocol.com'
         config.provider_url = 'https://provider.rinkeby.oceanprotocol.com'
```


github.com/AlgoveraAI/onshore/blob/main/notebooks/5-run-compute.ipynb

```
                 DATA_did,
                 compute_service.index,
                 ZERO_ADDRESS,
                 wallet,
                 dataset_order_requirements.computeAddress,
             )

In [32]:  # order & pay for algo
          algo_order_requirements = ocean.assets.order(
              ALG_did, wallet.address, service_type=algo_service.type
          )
          ALG_order_tx_id = ocean.assets.pay_for_service(
                  ocean.web3,
                  algo_order_requirements.amount,
                  algo_order_requirements.data_token_address,
                  ALG_did,
                  algo_service.index,
                  ZERO_ADDRESS,
                  wallet,
                  algo_order_requirements.computeAddress,
          )

In [33]:  compute_inputs = [ComputeInput(DATA_did, DATA_order_tx_id, compute_service.index)]

In [34]:  job_id = ocean.compute.start(
              compute_inputs,
              wallet,
              algorithm_did=ALG_did,
              algorithm_tx_id=ALG_order_tx_id,
              algorithm_data_token=alg_token.address
          )
          print(f"Started compute job with id: {job_id}")

         Started compute job with id: 94ed86622a6342178303dc4126e6c2d2

In [37]:  ocean.compute.status(DATA_did, job_id, wallet)

Out[37]: {'ok': False, 'status': 31, 'statusText': 'Data provisioning failed'}

In [ ]:   result = ocean.compute.result_file(DATA_did, job_id, 0, wallet)  # 0 index, means we retrieve the results from the first
```

Ocean Opportunities For SNI hackers

Create & monetize nature / climate datasets



docs.google.com/spreadsheets/d/1_OYV9yvf9p7_5NzD-u3W0pAnA7MT4D98DcJ

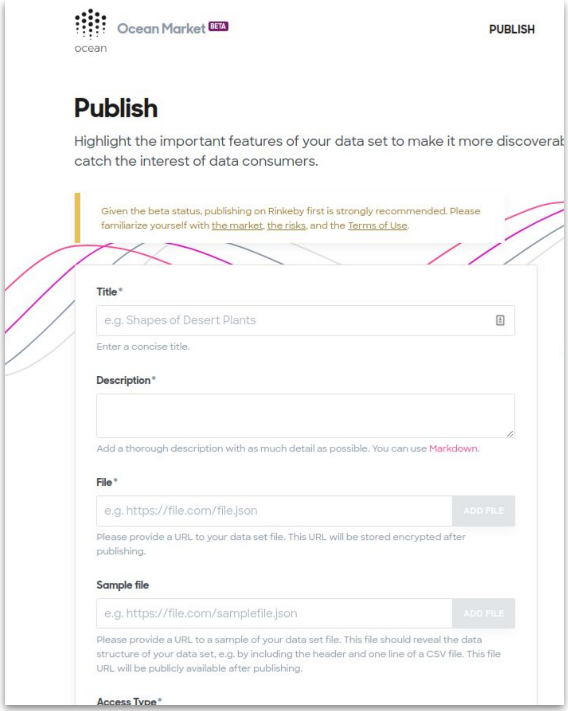
Untitled spreadsheet ☆

Your data!

Or your live datafeeds

Gsheet url

http or wss



Publish

Highlight the important features of your data set to make it more discoverab catch the interest of data consumers.

Given the beta status, publishing on Rinkeby first is strongly recommended. Please familiarize yourself with the market, the risks, and the Terms of Use.

Title*

e.g. Shapes of Desert Plants

Enter a concise title.

Description*

Add a thorough description with as much detail as possible. You can use Markdown.

File*

e.g. https://file.com/file.json

ADD FILE

Please provide a URL to your data set file. This URL will be stored encrypted after publishing.

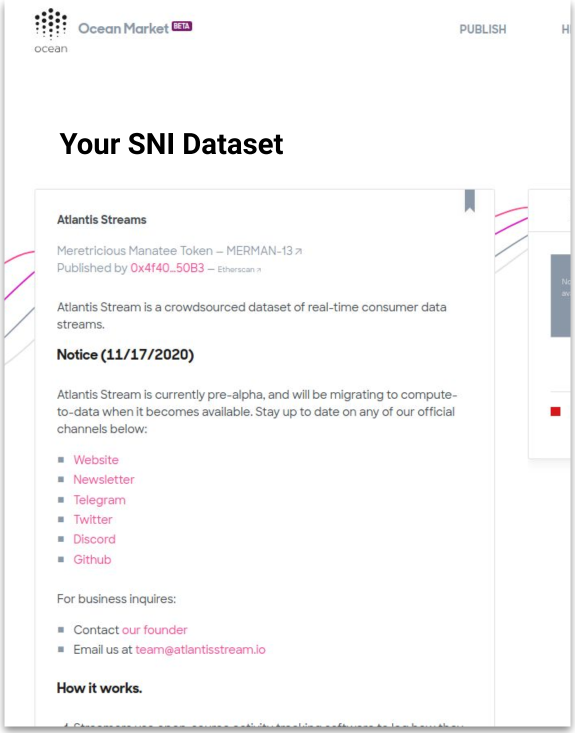
Sample file

e.g. https://file.com/samplefile.json

ADD FILE

Please provide a URL to a sample of your data set file. This file should reveal the data structure of your data set, e.g. by including the header and one line of a CSV file. This file URL will be publicly available after publishing.

Access Type*



Your SNI Dataset

Atlantis Streams

Meretricious Manatee Token — MERMAN-13 ↗
Published by 0x4f40...50B3 — Etherscan ↗

Atlantis Stream is a crowdsourced dataset of real-time consumer data streams.

Notice (11/17/2020)

Atlantis Stream is currently pre-alpha, and will be migrating to compute-to-data when it becomes available. Stay up to date on any of our official channels below:

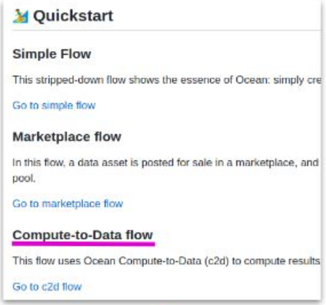
- Website
- Newsletter
- Telegram
- Twitter
- Discord
- Github

For business inquires:

- Contact our founder
- Email us at team@atlantisstream.io

How it works.

Programmatically sell ML models & predictions around nature / climate



Quickstart

Simple Flow

This stripped-down flow shows the essence of Ocean: simply cre

Go to simple flow

Marketplace flow

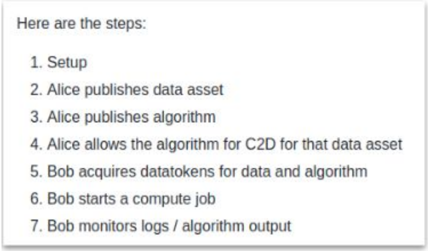
In this flow, a data asset is posted for sale in a marketplace, and pool.

Go to marketplace flow

Compute-to-Data flow

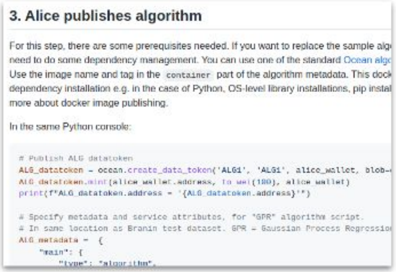
This flow uses Ocean Compute-to-Data (c2d) to compute results

Go to c2d flow



Here are the steps:

1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

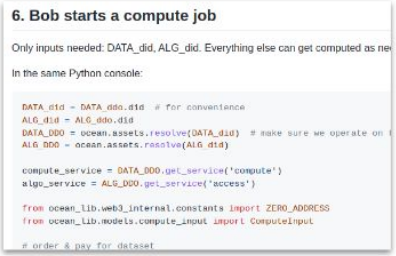


3. Alice publishes algorithm

For this step, there are some prerequisites needed. If you want to replace the sample alg
need to do some dependency management. You can use one of the standard Ocean algo
Use the image name and tag in the `container` part of the algorithm metadata. This dock
dependency installation e.g. in the case of Python, OS-level library installations, pip instal
more about docker image publishing.

In the same Python console:

```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob-
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regressio
ALG_metadata = {
    "main": {
        "type": "algorithm",
```

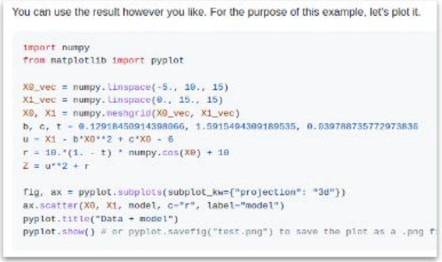


6. Bob starts a compute job

Only inputs needed: DATA_did, ALG_did. Everything else can get computed as ne

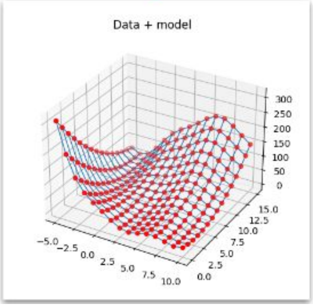
In the same Python console:

```
DATA_did = DATA_ddo.did  # for convenience
ALG_did = ALG_ddo.did
DATA_DDO = ocean.assets.resolve(DATA_did)  # make sure we operate on t
ALG_DDO = ocean.assets.resolve(ALG_did)

compute_service = DATA_DDO.get_service('compute')
algo_service = ALG_DDO.get_service('access')

from ocean_lib.web3_internal.constants import ZERO_ADDRESS
from ocean_lib.models.compute_input import ComputeInput

# order & pay for dataset
```



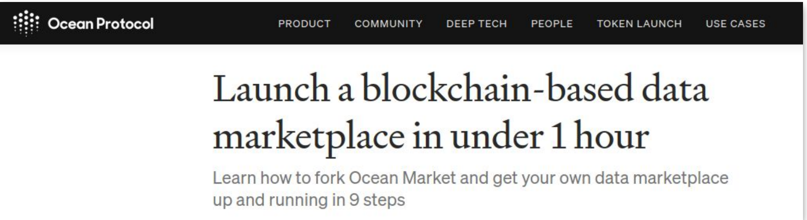
You can use the result however you like. For the purpose of this example, let's plot it.

```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.0397887357729738З6
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png f
```



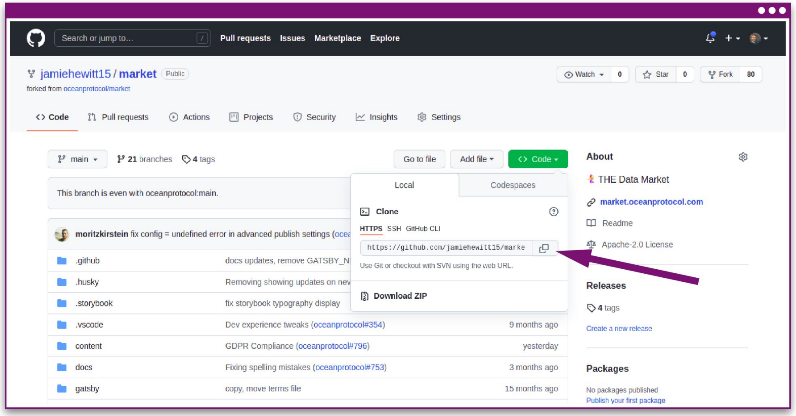
Create your own Nature / Climate data market

https://blog.oceanprotocol.com/launch-a-blockchain-based-data-marketplace-in-under-1-hour-9baa85a65ece



Launch a blockchain-based data marketplace in under 1 hour

Learn how to fork Ocean Market and get your own data marketplace up and running in 9 steps



jamiehewitt15 / market Public

forked from oceanprotocol/market

This branch is even with oceanprotocol:main.

moritzkirstein fix config = undefined error in advanced publish settings (ocea

.github docs updates, remove GATSBY_N

.husky Removing showing updates on nev

.storybook fix storybook typography display

.vscode Dev experience tweaks (oceanprotocol#354) 9 months ago

content GDPR Compliance (oceanprotocol#796) yesterday

docs Fixing spelling mistakes (oceanprotocol#753) 3 months ago

gatsby copy, move terms file 15 months ago

```
https://github.com/jamiehewitt15/marke
```

Use Git or checkout with SVN using the web URL.

About

THE Data Market

market.oceanprotocol.com

Readme

Apache-2.0 License

Releases

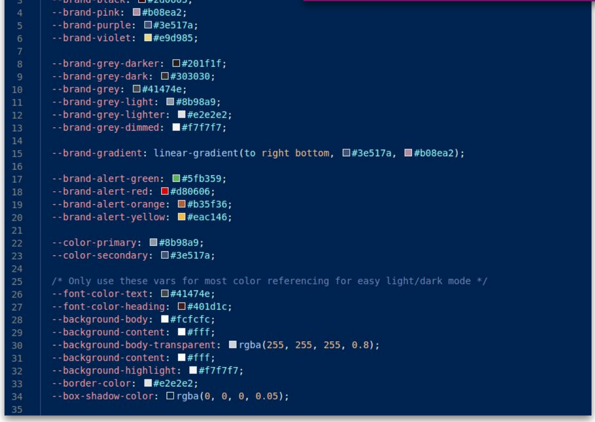
4 tags

Create a new release

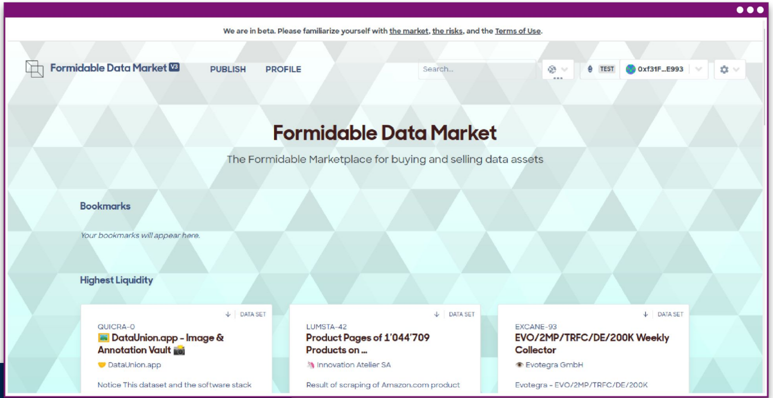
Packages

No packages published

Publish your first package



```
:root {
  --brand-white: #ffffff;
  --brand-black: #2d0605;
  --brand-pink: #b08ea2;
  --brand-purple: #3e517a;
  --brand-violet: #e9d985;

  --brand-grey-darker: #201f1f;
  --brand-grey-dark: #303030;
  --brand-grey: #41474e;
  --brand-grey-light: #8b98a9;
  --brand-grey-lighter: #e2e2e2;
  --brand-grey-dimmed: #f7f7f7;

  --brand-gradient: linear-gradient(to right bottom, #3e517a, #b08ea2);

  --brand-alert-green: #5fb359;
  --brand-alert-red: #d80606;
  --brand-alert-orange: #b35f36;
  --brand-alert-yellow: #eac146;

  --color-primary: #8b98a9;
  --color-secondary: #3e517a;

  /* Only use these vars for most color referencing for easy light/dark mode */
  --font-color-text: #41474e;
  --font-color-heading: #401d1c;
  --background-body: #fcfcfc;
  --background-content: #fff;
  --background-body-transparent: rgba(255, 255, 255, 0.8);
  --background-content: #fff;
  --background-highlight: #f7f7f7;
  --border-color: #e2e2e2;
  --box-shadow-color: rgba(0, 0, 0, 0.05);
```



We are in beta. Please familiarize yourself with the market, the risks, and the Terms of Use.

Formidable Data Market

The Formidable Marketplace for buying and selling data assets

Bookmarks

Your bookmarks will appear here.

Highest Liquidity

QUICRA-0
DataUnion.app - Image & Annotation Vault
DataUnion.app
Notice This dataset and the software stack

LUMSTA-42
Product Pages of 1'044'709 Products on ...
Innovation Atelier SA
Result of scraping of Amazon.com product

EXCANE-93
EVO/2MP/TRFC/DE/200K Weekly Collector
Evotegra GmbH
Evotegra - EVO/2MP/TRFC/DE/200K

Conclusion

ocean

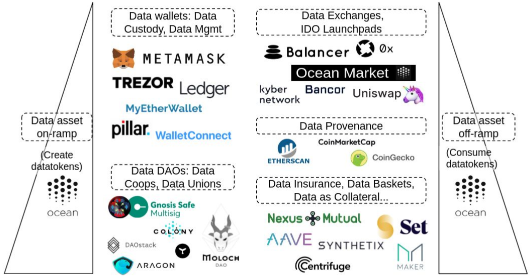
What is Ocean?

Ocean is...

1. **A community / ecosystem** of individuals and orgs driving to the mission (initiated by Ocean Protocol Foundation)
2. **A set of tools** as public infrastructure to facilitate the mission
3. **A token** (OCEAN) with incentives to grow & sustain the ecosystem

We are at the birth of an Open Data Economy

- For the first time ever, **data as an asset**, on open yet privacy-preserving data markets
- Leveraging the tools of crypto, from data wallets to DAOs to DEXes
- An ecosystem has formed. 100+ projects, dozens of collaborators big and small



Data asset on-ramp

(Create datatokens)

Data wallets: Data Custody, Data Mgmt

Data DAOs: Data Coops, Data Unions

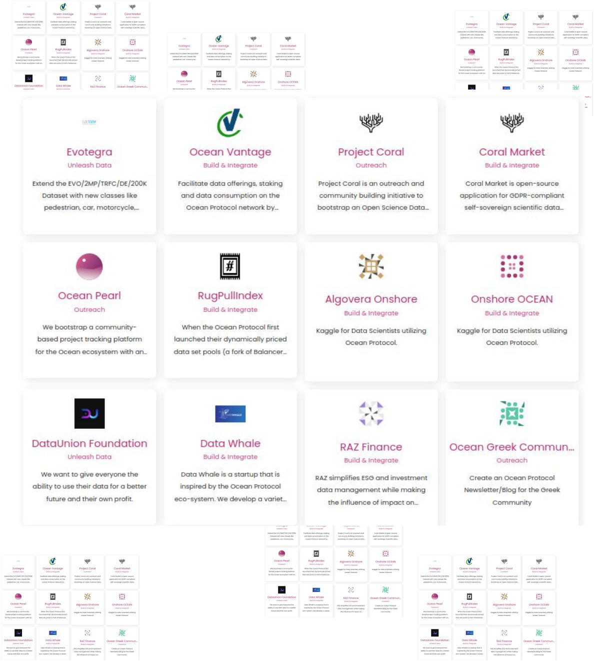
Data Exchanges, IDO Launchpads

Data Provenance

Data Insurance, Data Baskets, Data as Collateral...

Data asset off-ramp

(Consume datatokens)



Evotegra
Unleash Data
Extend the EVO/2MP/TRFC/DE/200K Dataset with new classes like pedestrian, car, motorcycle,...

Ocean Vantage
Build & Integrate
Facilitate data offerings, staking and data consumption on the Ocean Protocol network by...

Project Coral
Outreach
Project Coral is an outreach and community building initiative to bootstrap an Open Science Data...

Coral Market
Build & Integrate
Coral Market is open-source application for GDPR-compliant self-sovereign scientific data...

Ocean Pearl
Outreach
We bootstrap a community-based project tracking platform for the Ocean ecosystem with an...

RugPullIndex
Build & Integrate
When the Ocean Protocol first launched their dynamically priced data set pools (a fork of Balancer...

Algovera Onshore
Build & Integrate
Kaggle for Data Scientists utilizing Ocean Protocol.

Onshore OCEAN
Build & Integrate
Kaggle for Data Scientists utilizing Ocean Protocol.

DataUnion Foundation
Unleash Data
We want to give everyone the ability to use their data for a better future and their own profit.

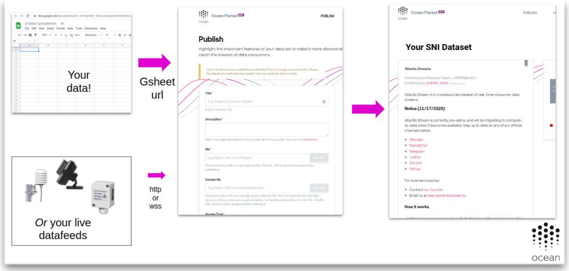
Data Whale
Build & Integrate
Data Whale is a startup that is inspired by the Ocean Protocol eco-system. We develop a variet...

RAZ Finance
Build & Integrate
RAZ simplifies ESG and investment data management while making the influence of impact on...

Ocean Greek Commun...
Outreach
Create an Ocean Protocol Newsletter/Blog for the Greek Community

There's opportunity for SNI hackers

Sell climate data, or ML models. Create your own market. Grants available!



Publish

Highlight the important features of your data set to make it more discoverab catch the interest of data consumers.

Title*

Description*

File*

Sample file

Access Type*

Your SNI Dataset

Atlantis Streams

Atlantis Stream is a crowdsourced dataset of real-time consumer data streams.

Notice (11/17/2020)

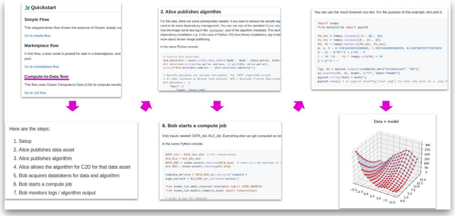
Atlantis Stream is currently pre-alpha, and will be migrating to compute-to-data when it becomes available. Stay up to date on any of our official channels below:

- Website
- Newsletter
- Telegram
- Twitter
- Discord
- Github

For business inquires:

- Contact our founder
- Email us at team@atlantisstream.io

How it works.



Quickstart

Simple Flow

This stripped-down flow shows the essence of Ocean: simply cre

Go to simple flow

Marketplace flow

In this flow, a data asset is posted for sale in a marketplace, and pool.

Go to marketplace flow

Compute-to-Data flow

This flow uses Ocean Compute-to-Data (c2d) to compute results

Go to c2d flow

Here are the steps:

1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

3. Alice publishes algorithm

For this step, there are some prerequisites needed. If you want to replace the sample alg need to do some dependency management. You can use one of the standard Ocean alg Use the image name and tag in the `container` part of the algorithm metadata. This dock dependency installation e.g. in the case of Python, OS-level library installations, pip instal more about docker image publishing.

In the same Python console:

```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob=
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regressio
ALG_metadata = {
    "main": {
        "type": "algorithm",
```

6. Bob starts a compute job

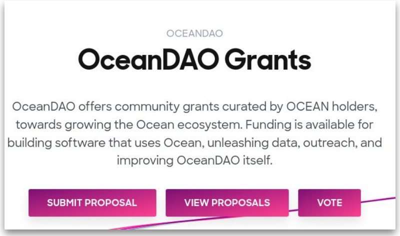
Only inputs needed: DATA_did, ALG_did. Everything else can get computed as ne

In the same Python console:

```
DATA_did = DATA_ddo.did  # for convenience
ALG_did = ALG_ddo.did
DATA_DDO = ocean.assets.resolve(DATA_did)  # make sure we operate on t
ALG_DDO = ocean.assets.resolve(ALG_did)

compute_service = DATA_DDO.get_service('compute')
algo_service = ALG_DDO.get_service('access')

from ocean_lib.web3_internal.constants import ZERO_ADDRESS
from ocean_lib.models.compute_input import ComputeInput

# order & pay for dataset
```

You can use the result however you like. For the purpose of this example, let's plot it.

```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.039788735772973836
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png f
```

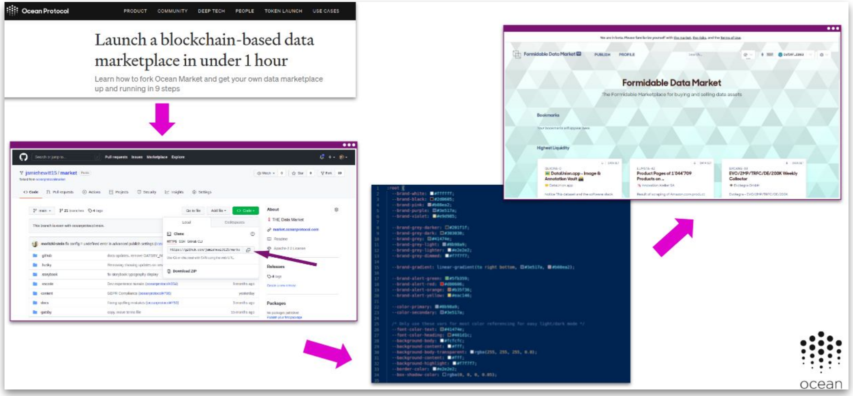


OCEANDAO

OceanDAO Grants

OceanDAO offers community grants curated by OCEAN holders, towards growing the Ocean ecosystem. Funding is available for building software that uses Ocean, unleashing data, outreach, and improving OceanDAO itself.

https://oceanprotocol.com/dao



Launch a blockchain-based data marketplace in under 1 hour

Learn how to fork Ocean Market and get your own data marketplace up and running in 9 steps

Formidable Data Market

The Formidable Marketplace for buying and selling data assets

Trent McConaghy

@trentmc0

Resources

Site	oceanprotocol.com
Code	github.com/oceanprotocol
Chat	discord.com/invite/TnXjkR5

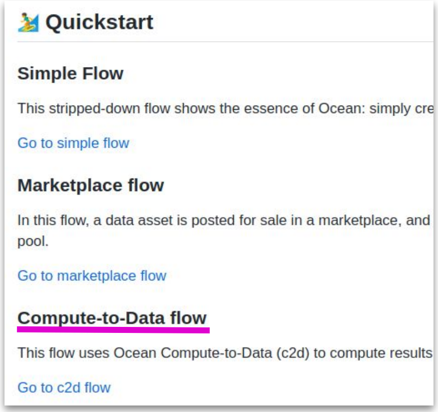
ocean

Appendix:
C2D Quickstart Details

ocean

C2D Quickstart via Ocean.py: Overview

github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



Quickstart

Simple Flow

This stripped-down flow shows the essence of Ocean: simply cre

Go to simple flow

Marketplace flow

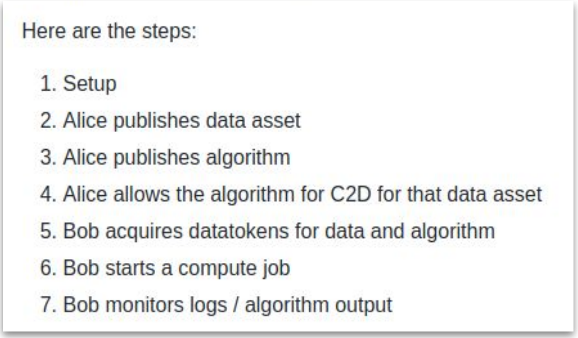
In this flow, a data asset is posted for sale in a marketplace, and pool.

Go to marketplace flow

Compute-to-Data flow

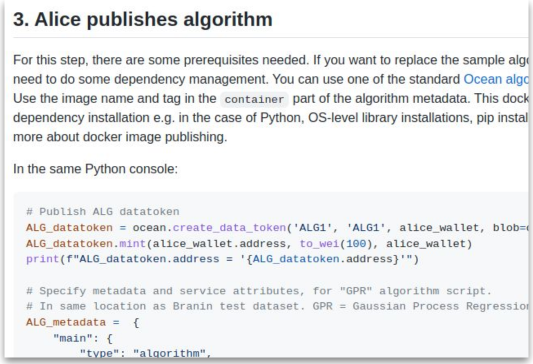
This flow uses Ocean Compute-to-Data (c2d) to compute results

Go to c2d flow



Here are the steps:

1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

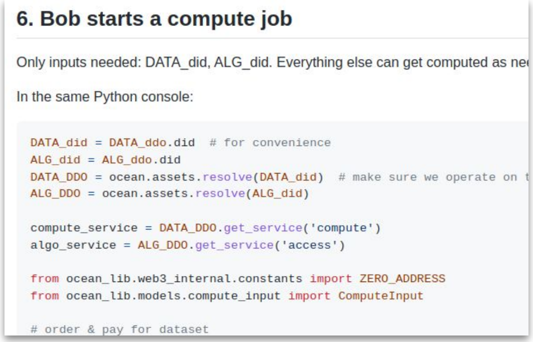


3. Alice publishes algorithm

For this step, there are some prerequisites needed. If you want to replace the sample alg need to do some dependency management. You can use one of the standard Ocean algo Use the image name and tag in the `container` part of the algorithm metadata. This dock dependency installation e.g. in the case of Python, OS-level library installations, pip instal more about docker image publishing.

In the same Python console:

```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob=
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regressio
ALG_metadata =  {
    "main": {
        "type": "algorithm",
```

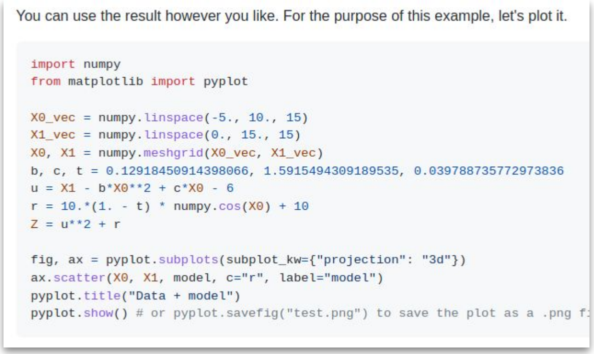


6. Bob starts a compute job

Only inputs needed: DATA_did, ALG_did. Everything else can get computed as ne

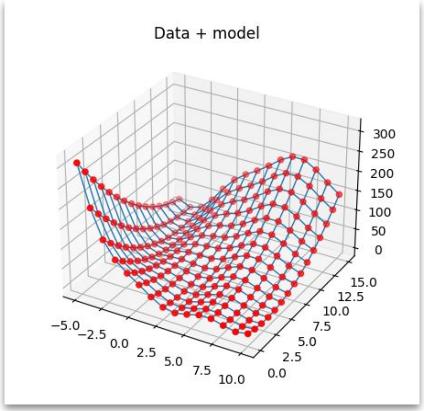
In the same Python console:

```
DATA_did = DATA_ddo.did  # for convenience
ALG_did = ALG_ddo.did
DATA_DDO = ocean.assets.resolve(DATA_did)  # make sure we operate on t
ALG_DDO = ocean.assets.resolve(ALG_did)

compute_service = DATA_DDO.get_service('compute')
algo_service = ALG_DDO.get_service('access')

from ocean_lib.web3_internal.constants import ZERO_ADDRESS
from ocean_lib.models.compute_input import ComputeInput

# order & pay for dataset
```



You can use the result however you like. For the purpose of this example, let's plot it.

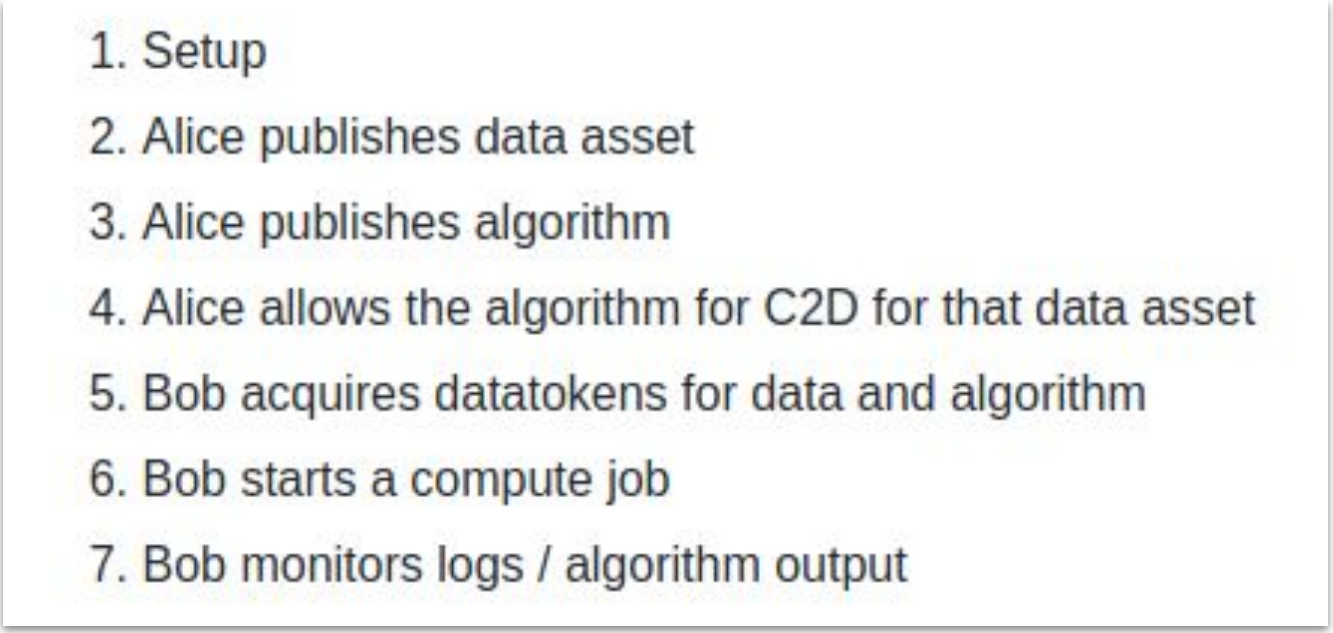
```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.039788735772973836
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png f
```



Data + model

C2D Quickstart: Steps

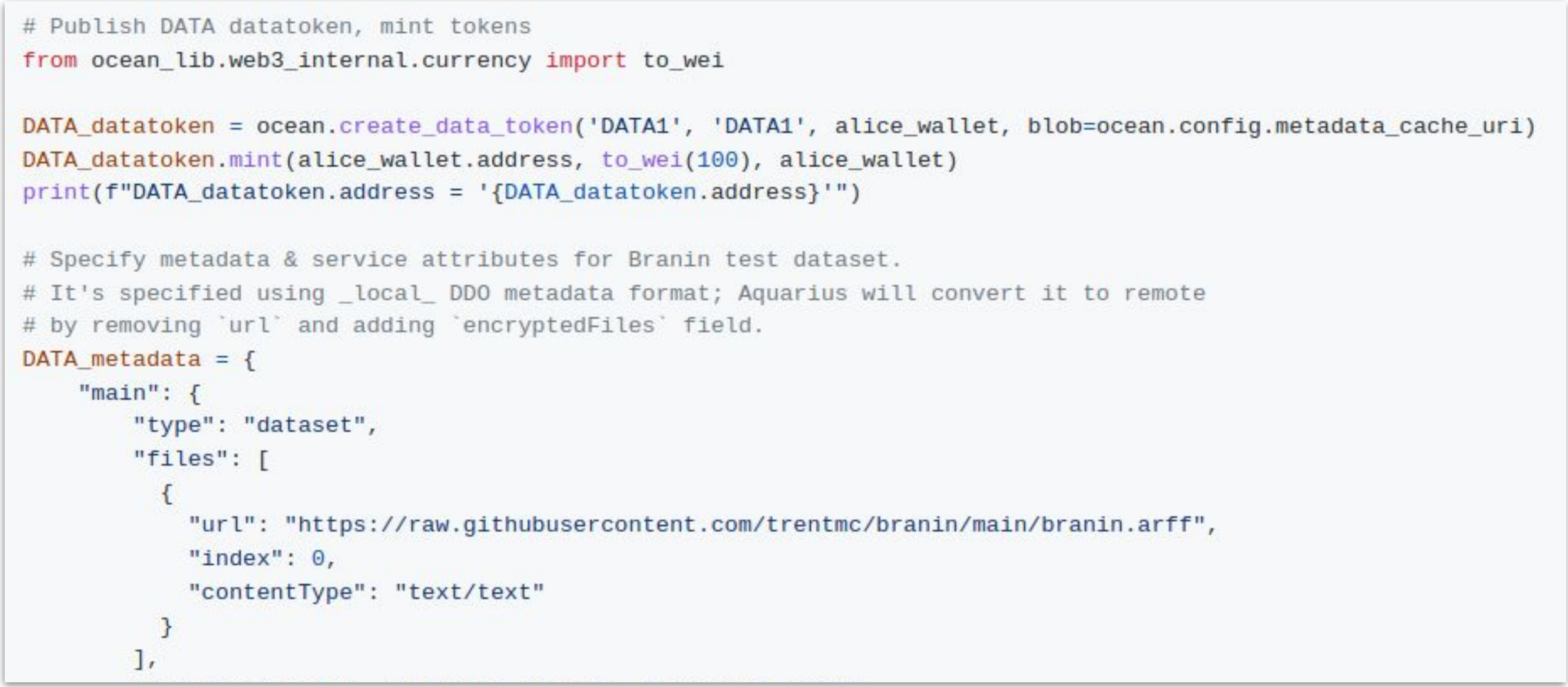
github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

C2D Quickstart: Step 2: Publish dataset

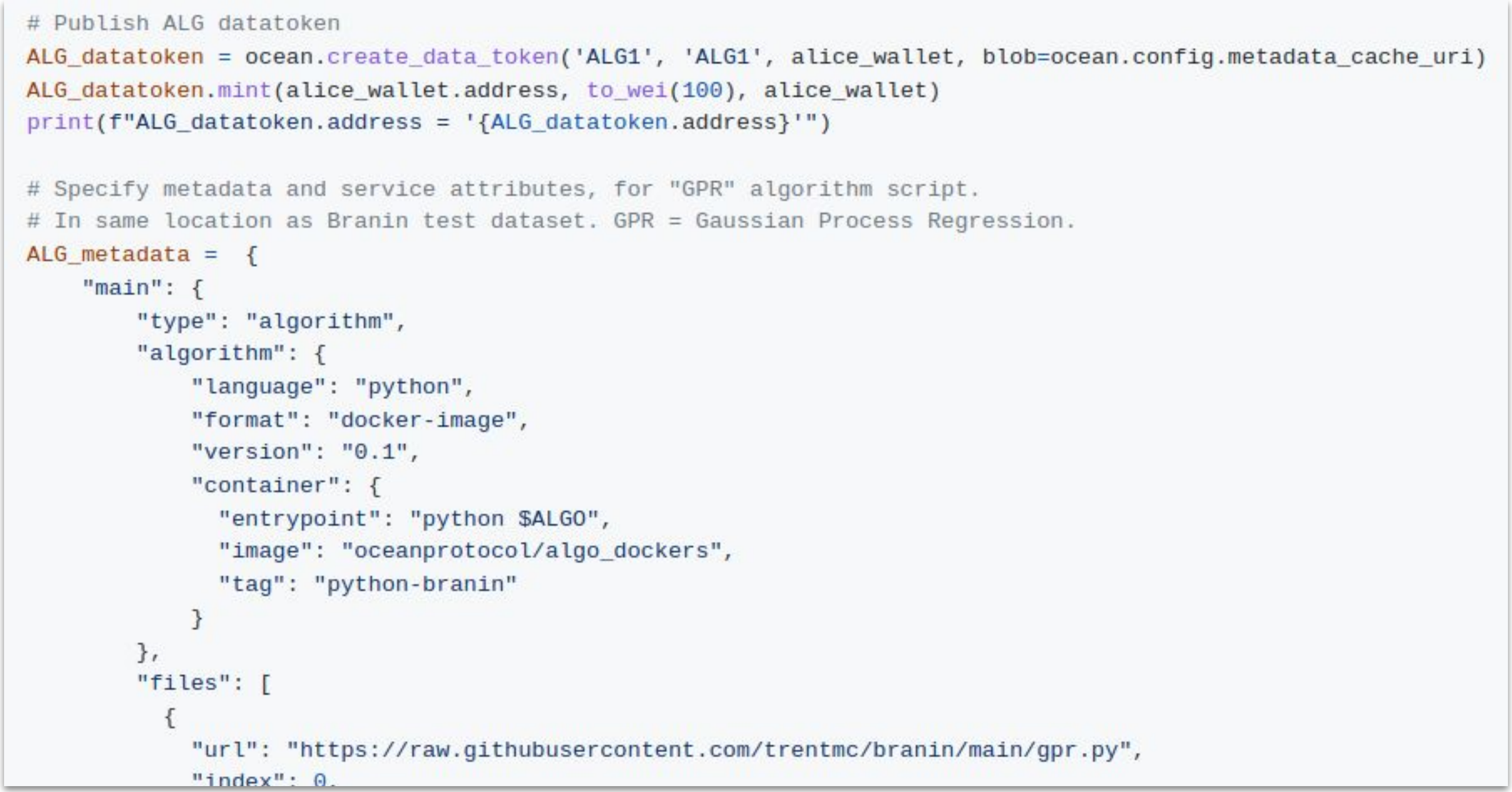
github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



```
# Publish DATA datatoken, mint tokens
from ocean_lib.web3_internal.currency import to_wei

DATA_datatoken = ocean.create_data_token('DATA1', 'DATA1', alice_wallet, blob=ocean.config.metadata_cache_uri)
DATA_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"DATA_datatoken.address = '{DATA_datatoken.address}'")

# Specify metadata & service attributes for Branin test dataset.
# It's specified using _local_ DDO metadata format; Aquarius will convert it to remote
# by removing `url` and adding `encryptedFiles` field.
DATA_metadata = {
    "main": {
        "type": "dataset",
        "files": [
          {
            "url": "https://raw.githubusercontent.com/trentmc/branin/main/branin.arff",
            "index": 0,
            "contentType": "text/text"
          }
        ],
```

C2D Quickstart: Step 3: Publish algorithm

github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob=ocean.config.metadata_cache_uri)
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regression.
ALG_metadata =  {
    "main": {
        "type": "algorithm",
        "algorithm": {
            "language": "python",
            "format": "docker-image",
            "version": "0.1",
            "container": {
              "entrypoint": "python $ALGO",
              "image": "oceanprotocol/algo_dockers",
              "tag": "python-branin"
            }
        },
        "files": [
          {
            "url": "https://raw.githubusercontent.com/trentmc/branin/main/gpr.py",
            "index": 0,
```

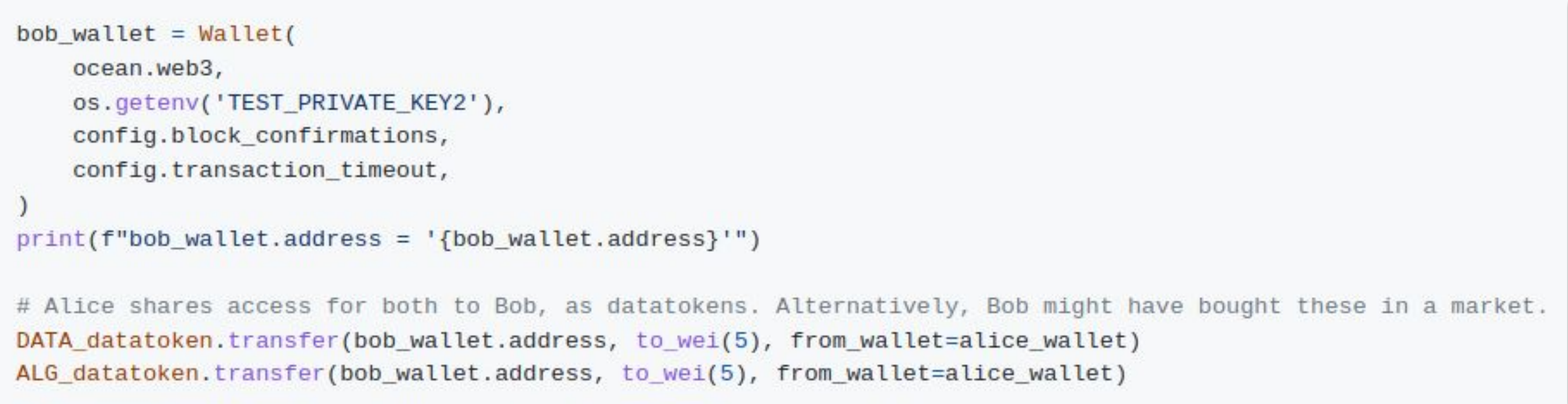
C2D Quickstart: Step 4: dataset allows algorithm

github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md

```
from ocean_lib.assets.trusted_algorithms import add_publisher_trusted_algorithm
add_publisher_trusted_algorithm(DATA_ddo, ALG_ddo.did, config.metadata_cache_uri)
ocean.assets.update(DATA_ddo, publisher_wallet=alice_wallet)
```

C2D Quickstart: Step 5: get data & alg assets

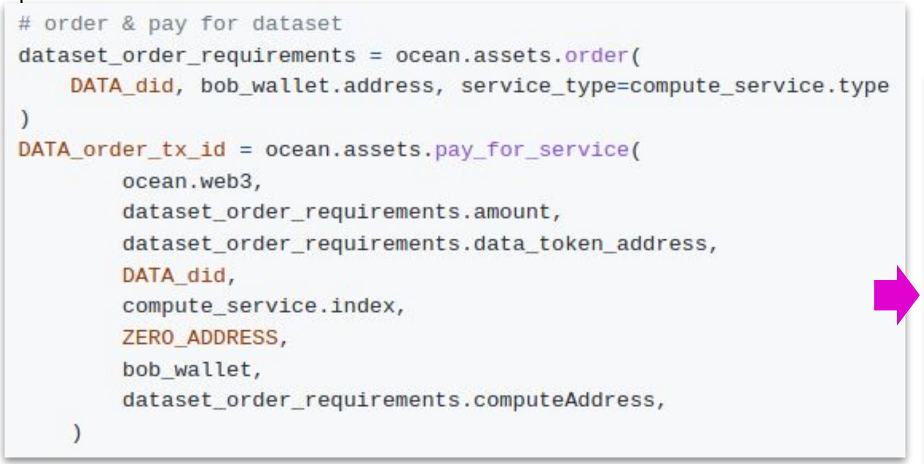
github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



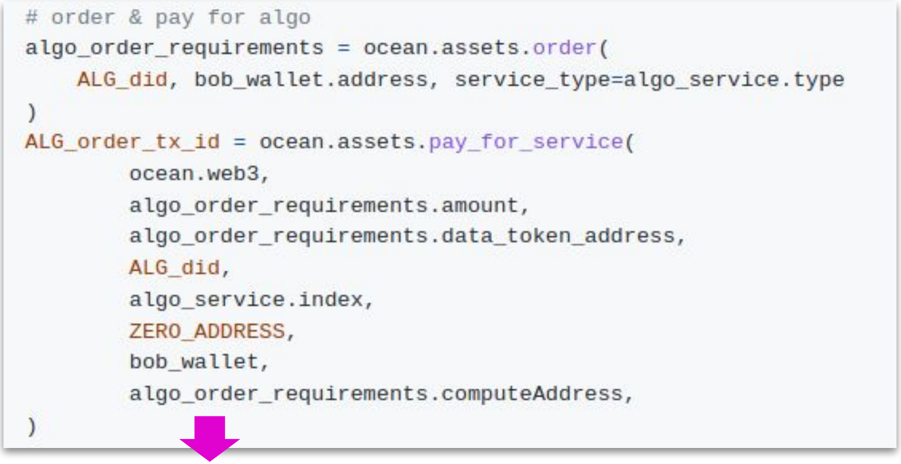
```
bob_wallet = Wallet(
    ocean.web3,
    os.getenv('TEST_PRIVATE_KEY2'),
    config.block_confirmations,
    config.transaction_timeout,
)
print(f"bob_wallet.address = '{bob_wallet.address}'")

# Alice shares access for both to Bob, as datatokens. Alternatively, Bob might have bought these in a market.
DATA_datatoken.transfer(bob_wallet.address, to_wei(5), from_wallet=alice_wallet)
ALG_datatoken.transfer(bob_wallet.address, to_wei(5), from_wallet=alice_wallet)
```

C2D Quickstart: Step 6: start compute

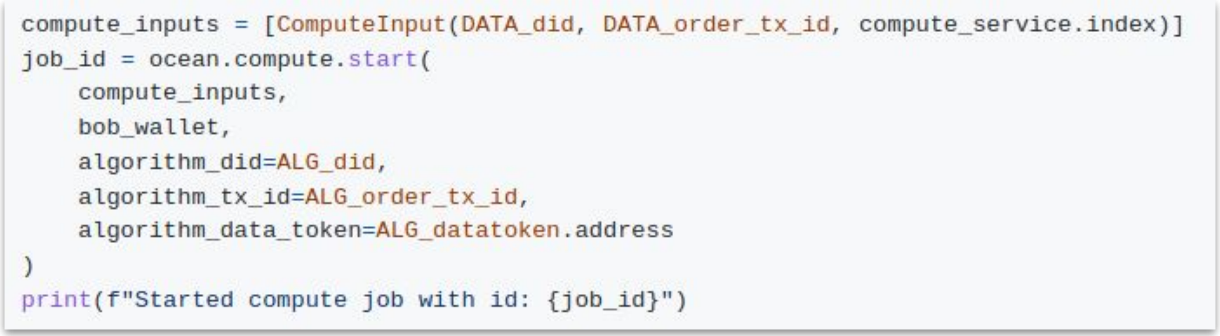
github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



```
# order & pay for dataset
dataset_order_requirements = ocean.assets.order(
    DATA_did, bob_wallet.address, service_type=compute_service.type
)
DATA_order_tx_id = ocean.assets.pay_for_service(
        ocean.web3,
        dataset_order_requirements.amount,
        dataset_order_requirements.data_token_address,
        DATA_did,
        compute_service.index,
        ZERO_ADDRESS,
        bob_wallet,
        dataset_order_requirements.computeAddress,
    )
```



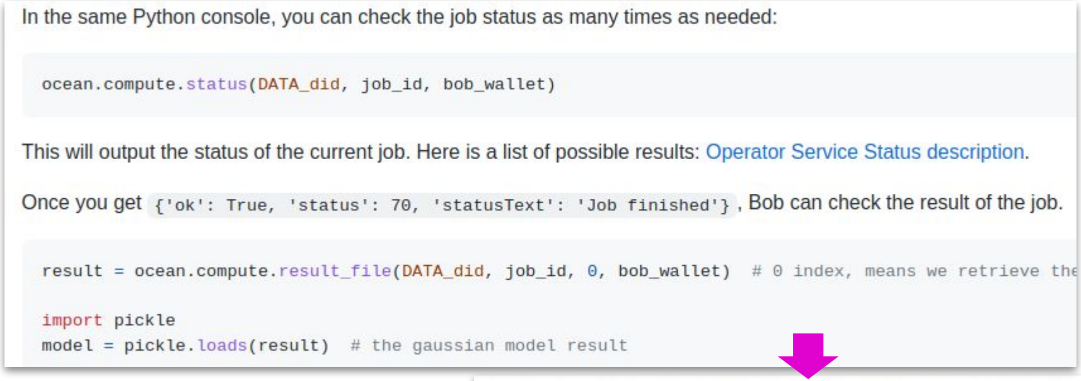
```
# order & pay for algo
algo_order_requirements = ocean.assets.order(
    ALG_did, bob_wallet.address, service_type=algo_service.type
)
ALG_order_tx_id = ocean.assets.pay_for_service(
        ocean.web3,
        algo_order_requirements.amount,
        algo_order_requirements.data_token_address,
        ALG_did,
        algo_service.index,
        ZERO_ADDRESS,
        bob_wallet,
        algo_order_requirements.computeAddress,
)
```



```
compute_inputs = [ComputeInput(DATA_did, DATA_order_tx_id, compute_service.index)]
job_id = ocean.compute.start(
    compute_inputs,
    bob_wallet,
    algorithm_did=ALG_did,
    algorithm_tx_id=ALG_order_tx_id,
    algorithm_data_token=ALG_datatoken.address
)
print(f"Started compute job with id: {job_id}")
```

C2D Quickstart: step 7: see output

github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



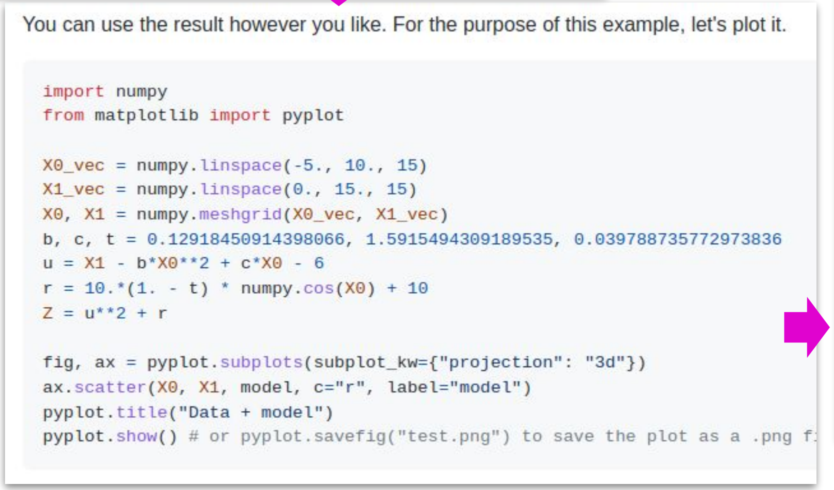
In the same Python console, you can check the job status as many times as needed:

```
ocean.compute.status(DATA_did, job_id, bob_wallet)
```

This will output the status of the current job. Here is a list of possible results: Operator Service Status description.

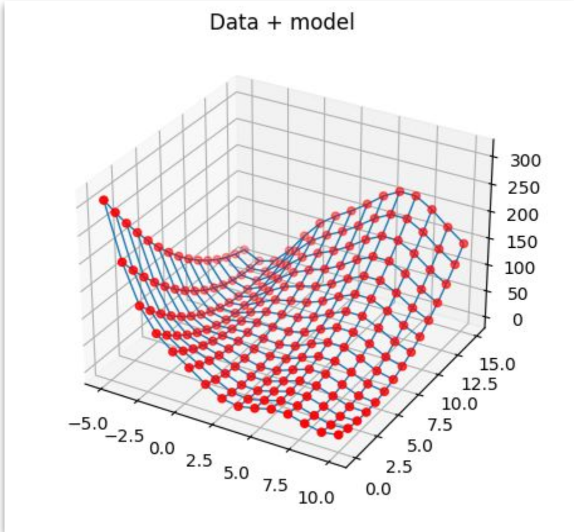
Once you get `{'ok': True, 'status': 70, 'statusText': 'Job finished'}`, Bob can check the result of the job.

```
result = ocean.compute.result_file(DATA_did, job_id, 0, bob_wallet)  # 0 index, means we retrieve the

import pickle
model = pickle.loads(result)  # the gaussian model result
```



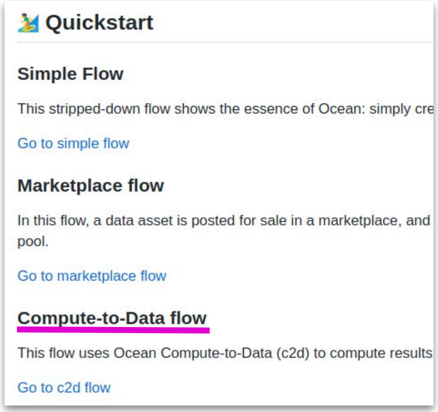
You can use the result however you like. For the purpose of this example, let's plot it.

```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.039788735772973836
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png f
```



C2D Quickstart via Ocean.py: Recap

github.com/oceanprotocol/ocean.py/blob/main/READMEs/c2d-flow.md



Quickstart

Simple Flow

This stripped-down flow shows the essence of Ocean: simply cre

Go to simple flow

Marketplace flow

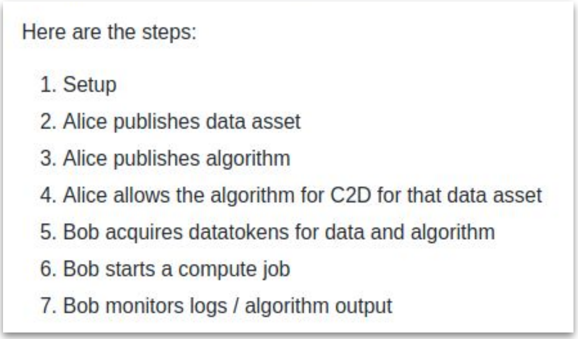
In this flow, a data asset is posted for sale in a marketplace, and pool.

Go to marketplace flow

Compute-to-Data flow

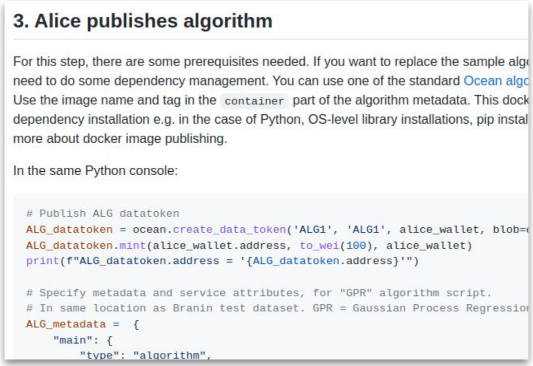
This flow uses Ocean Compute-to-Data (c2d) to compute results

Go to c2d flow



Here are the steps:

1. Setup
2. Alice publishes data asset
3. Alice publishes algorithm
4. Alice allows the algorithm for C2D for that data asset
5. Bob acquires datatokens for data and algorithm
6. Bob starts a compute job
7. Bob monitors logs / algorithm output

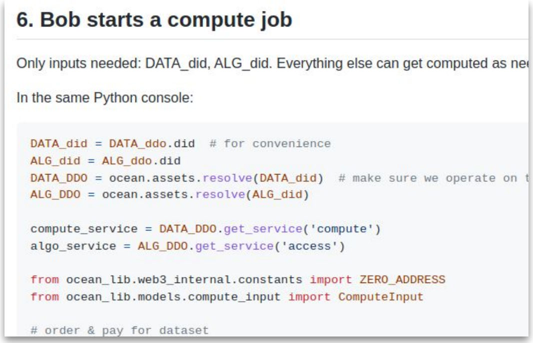


3. Alice publishes algorithm

For this step, there are some prerequisites needed. If you want to replace the sample alg
need to do some dependency management. You can use one of the standard Ocean algo
Use the image name and tag in the `container` part of the algorithm metadata. This dock
dependency installation e.g. in the case of Python, OS-level library installations, pip instal
more about docker image publishing.

In the same Python console:

```
# Publish ALG datatoken
ALG_datatoken = ocean.create_data_token('ALG1', 'ALG1', alice_wallet, blob=
ALG_datatoken.mint(alice_wallet.address, to_wei(100), alice_wallet)
print(f"ALG_datatoken.address = '{ALG_datatoken.address}'")

# Specify metadata and service attributes, for "GPR" algorithm script.
# In same location as Branin test dataset. GPR = Gaussian Process Regressio
ALG_metadata =  {
    "main": {
        "type": "algorithm",
```

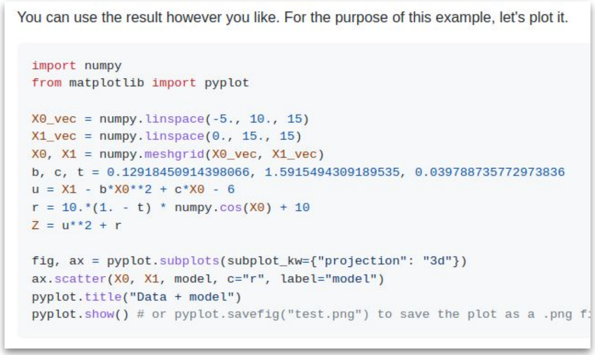


6. Bob starts a compute job

Only inputs needed: DATA_did, ALG_did. Everything else can get computed as ne

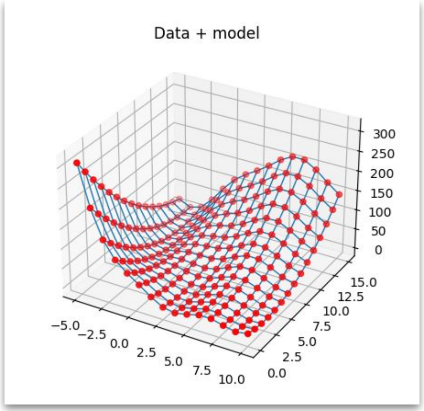
In the same Python console:

```
DATA_did = DATA_ddo.did  # for convenience
ALG_did = ALG_ddo.did
DATA_DDO = ocean.assets.resolve(DATA_did)  # make sure we operate on t
ALG_DDO = ocean.assets.resolve(ALG_did)

compute_service = DATA_DDO.get_service('compute')
algo_service = ALG_DDO.get_service('access')

from ocean_lib.web3_internal.constants import ZERO_ADDRESS
from ocean_lib.models.compute_input import ComputeInput

# order & pay for dataset
```



You can use the result however you like. For the purpose of this example, let's plot it.

```
import numpy
from matplotlib import pyplot

X0_vec = numpy.linspace(-5., 10., 15)
X1_vec = numpy.linspace(0., 15., 15)
X0, X1 = numpy.meshgrid(X0_vec, X1_vec)
b, c, t = 0.12918450914398066, 1.5915494309189535, 0.0397887357729738 36
u = X1 - b*X0**2 + c*X0 - 6
r = 10.*(1. - t) * numpy.cos(X0) + 10
Z = u**2 + r

fig, ax = pyplot.subplots(subplot_kw={"projection": "3d"})
ax.scatter(X0, X1, model, c="r", label="model")
pyplot.title("Data + model")
pyplot.show() # or pyplot.savefig("test.png") to save the plot as a .png fi
```


Data + model