

Lesson 14

Today's Lesson - Anchor / NFTs / DeFi

Anchor Quick Start

Assumes node / npm / yarn installed

Rust

```
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

Solana CLI

```
sh -c "$(curl -sSfL https://release.solana.com/v1.14.9/install)"
```

Create wallet

```
mkdir ~/my-solana-wallet  
solana-keygen new --outfile ~/my-solana-wallet/my-keypair.json
```

Install Anchor

```
cargo install --git https://github.com/project-serum/anchor avm --locked --force  
  
avm install 0.25.0  
avm use latest  
  
anchor --version
```

To set up a project

```
anchor init <project-name>
```

Build the project with

```
anchor build
```

Remember you will need to change the declare_id , to do this you can run

```
anchor keys list
```

then add the key to the declare_id! macro in your lib.rs
and re run

```
anchor build
```

For testing run

```
anchor test
```

The testing will be done on a local cluster, once you are happy you can deploy to devnet.

Change the `provider.cluster` variable in `Anchor.toml` to `devnet`.

```
[provider]
cluster = "devnet"
wallet = "/home/extropy/.config/solana/id.json"
```

then run

```
anchor deploy
```

Useful Account macro attributes

Full list [here](#)

Common useful ones

- `#[account(mut)]` - Used to tell Solana you will change the data in the account and it will need to serialise this back.
- `#[account(signer)]` verifies that an account is also the signer.
- `#[account(has_one=target)]` this uses a constraint to ensure that a field within the account matches what we have in the account structure.
- `#[account(owner=target)]` Similar to the previous example but ensures that owner matches the target.

Anchor Program Registry

See [Docs](#)



Search by name or address

⌘K

The Anchor Program Registry allows you to publish the source code of your programs.

Process

1. You need to create an account at [apr.dev](#)
2. Create an token, from the account section on the top right
3. Connect your anchor session

```
anchor login <token>
```

4. Prepare the program details

You need to configure your anchor.toml file

It will look something like this

```
anchor_version = "0.25.0"

[workspace]
members = ["programs/multisig"]

[provider]
cluster = "mainnet"
wallet = "~/.config/solana/id.json"

[programs.mainnet]
multisig = "A9HAbnCwoD6f2NkZobKFf6buJoN9gUVVvX5PoUnDHS6u"
```

```
[programs.localnet]
multisig = "A9HAbnCwoD6f2NkZobKFf6buJoN9gUVVvX5PoUnDHS6u"
```

5. To publish the details to the program registry run

```
anchor publish <program name>
```

Solana Mobile Stack

This was launched in June this year alongside Saga, an Android mobile phone with features tightly integrated with Solana



The Solana Mobile Stack (SMS) provides a new set of libraries for wallets and apps, allowing developers to create rich mobile experiences on Solana, the world's most performant blockchain, and is built to run alongside Android. The SDK provides libraries and programming interfaces for Android apps and secure key private storage, simplifying the developer experience to build and extend dApps functionality for Solana.

The Solana dApp Store will provide a distribution channel for apps that want to establish direct relationships with their customers, allowing them to transact without platform fees. The Solana dApp Store will co-exist with Google Play on devices, providing access to web2 and web3 apps.

To sync your phantom and file system wallet

export your 12 word phrase from phantom

```
solana-keygen recover --force 'prompt://?key=0/0' -o <keypair.json>
```

NFTs

Welcoming Creators & Builders to Solana

Metaplex makes it easy to build your project and grow your community in the world's largest NFT ecosystem.

[Metaplex for Creators](#)

[Explore Protocol](#)



NFT Overview

NFTs are Non Fungible Tokens

They can be used for many things, but commonly they are used to tokenise or establish ownership of digital assets.

Although the standard was established in 2018, their popularity increased greatly in 2020 / 2021 and they became a mainstay of DeFi.

Non fungible means that each token is unique, unlike SOL for example for any one SOL is equivalent to any other.

An NFT therefore will have a means to identify it, we can then find who owns a particular NFT.

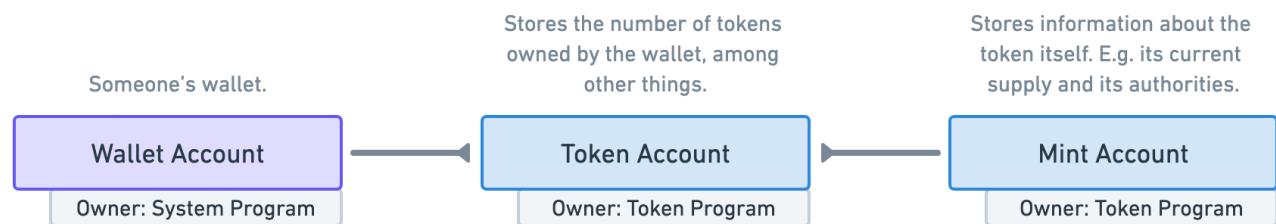
When used with digital assets, such as images the NFT will include a link to the asset.

Typically the assets are stored in decentralised storage systems such as Arweave, IPFS, NFT.Storage etc. The details of the location of the asset are often referred to as the metadata of the NFT.

More recently there has become a need for semi fungible tokens, these are SPL tokens with a supply greater than 1 but which has typical NFT attributes such as an image.

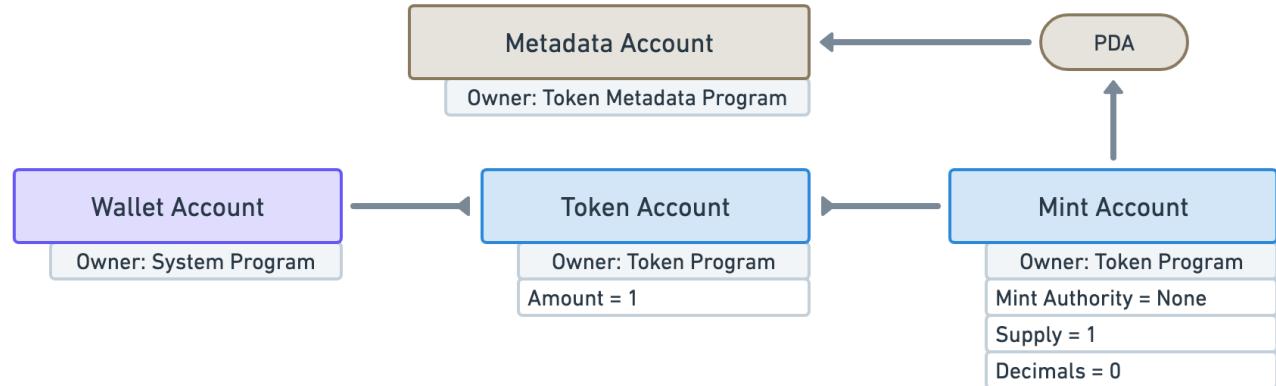
NFTs on Solana

See [Documentation](#)

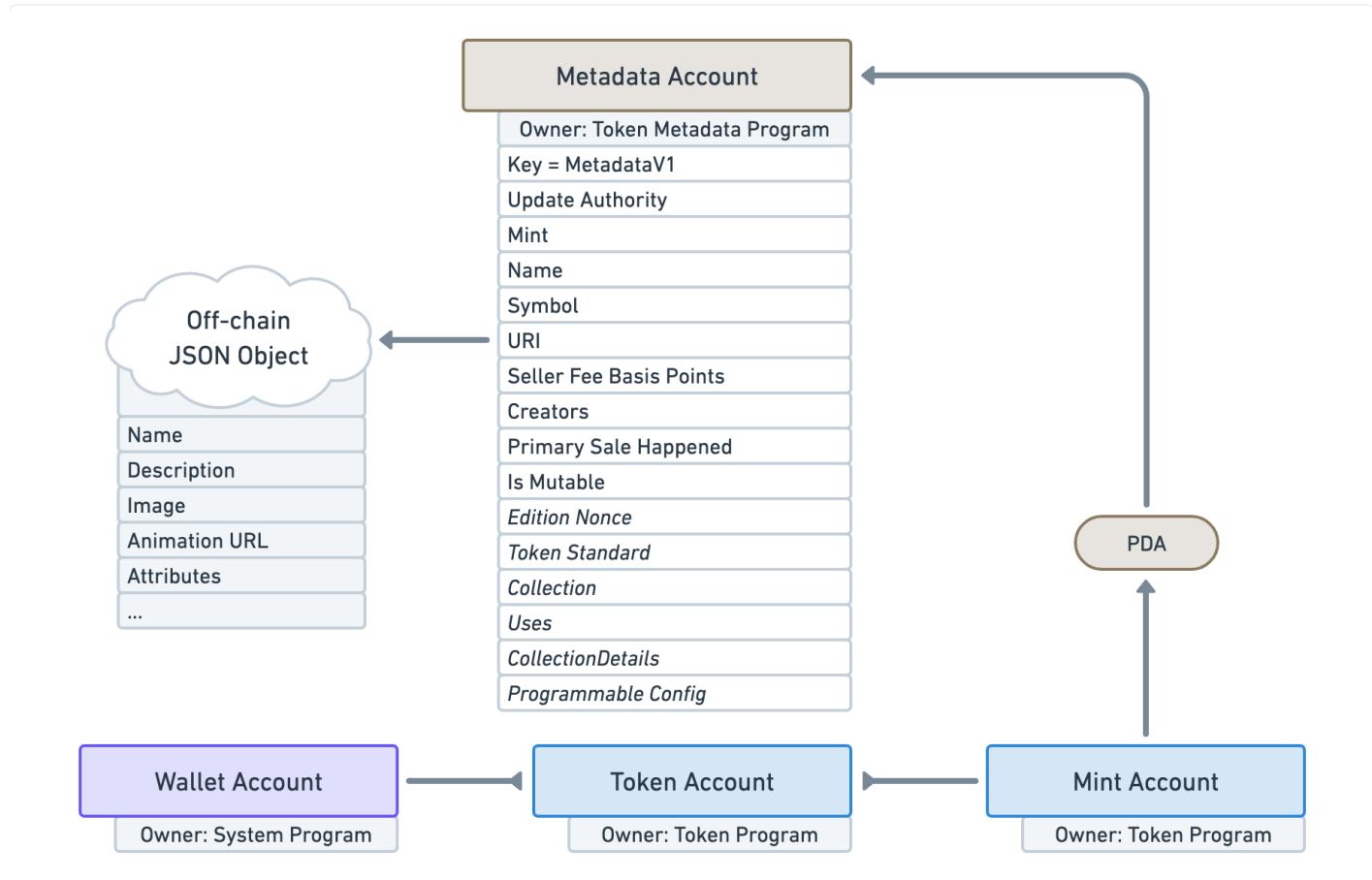
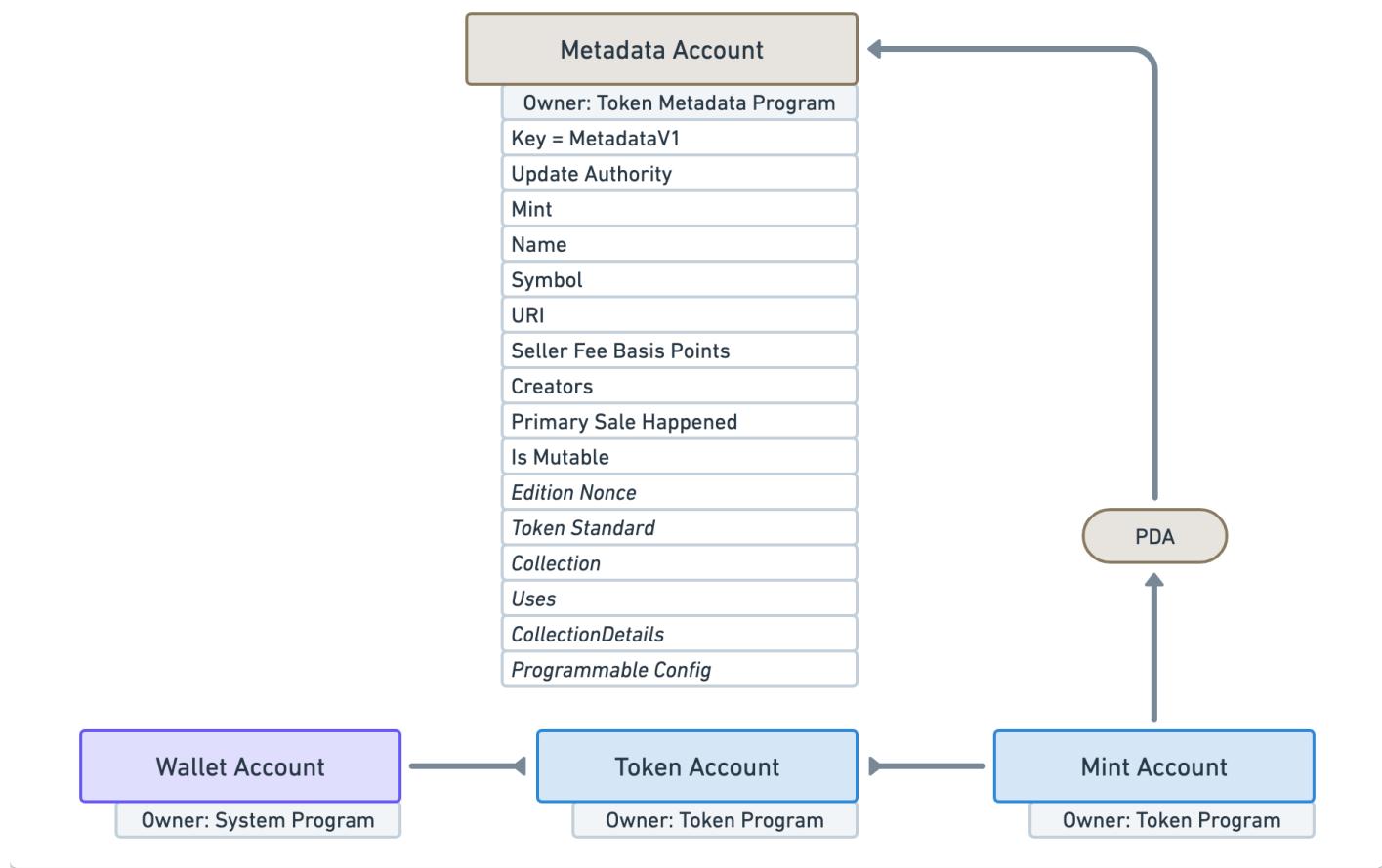


More precisely, NFTs in Solana are Mint Accounts with the following characteristics:

- It has **a supply of 1**, meaning only one token is in circulation.
- It has **zero decimals**, meaning there cannot be such a thing as 0.5 tokens.
- It has **no mint authority**, meaning no one can ever mint additional tokens.



Token Metadata



Metaplex overview

[Metaplex](#) is a framework for NFT launches, sales, auctions and partitioning that includes on-chain programs as well as client libraries. Some of the programs are not stable while others do not have a ready client.

Metaplex has 3 main projects

- [Token Metadata](#) - the NFT standard for Solana
- [Candy Machine v3™](#) - a Profile Picture (PFP) focused tool that works like the gumball-style candy machines of old
- [Auction House](#) - a decentralized sales protocol for NFT marketplaces

Metaplex NFTs off-chain data can be stored on either Arweave, IPFS or AWS and multiple file formats are supported.

Wallets and websites that support Metaplex will be able to display all or some of these formats.

File Type	Extension
Audio	.mp3, .flac, .wav
Img	.png, .jpg, .gif
Movie	.mp4, .mov, .webm
VR	.gib
HTML	.html

Collections can be derived from existing collections.

Reasons to use Metaplex

A creator that is agnostic to the platform that will host NFTs, may consider Metaplex/Solana due to cheap gas costs.

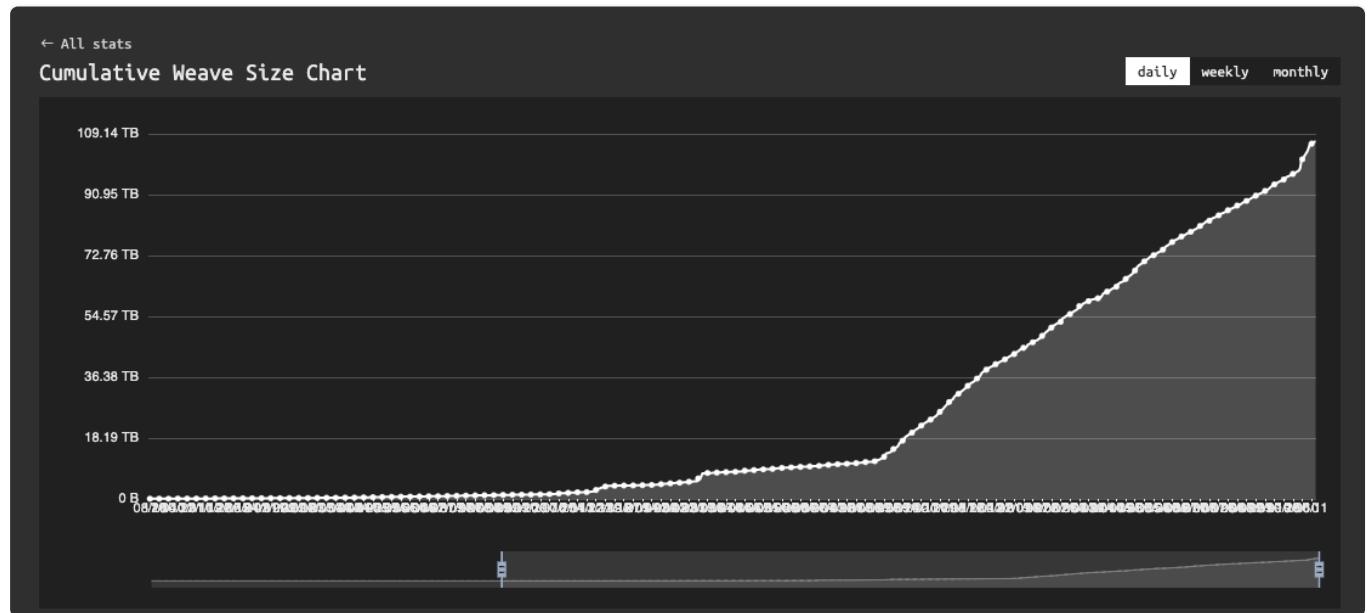
The price of a single NFT mint varies due to the size of off-chain data and SOL price but will usually be under \$3.

Arweave

Features

- Decentralised storage built on top of the Arweave protocol
- Permanent and resilient
- Proof of Access mechanism (similar to Filecoin's Proof of Space and Proof of Replication)
- Incentivises quick retrieval
- Data replicated and distributed across the network

Currently adding approx 6 TB per month



Storage details

- Mechanism to distribute tokens to sustainably incentivise perpetual storage or arbitrary quantities of data.
- Based on a logarithmically decreasing \$/GB-h.
- Storage media assumed to take 434 yrs to reach max theoretical data density limit of 1.53×10^{67} bits/cm³ (currently 1.66×10^{12})
- Storage medium reliability increases.
- Incentive for cheap storage grows as humanity's demand for data is growing.
- Predictions for future safekeeping of data based on ultra-low cost of storage medium.
- Arweave expected to be 'nested' inside future storage systems.

Chain Architecture

- Block constructed from previous block, recall block and transactions
- Recall bloc
 - A deterministic but unpredictable choice of block from the "weave's" history
 - Prevents data from being lost if it is rarely accessed
 - Lesser known blocks increase miner fees

Metaplex architecture

Metaplex

A top level manager that knows the existence of other programs and interacts with them via CPI to accomplish necessary logic.

Token Metadata

This program allows decoration of SPL tokens with PDAs that store additional on-chain data and in them a pointer to the off-chain data.

Token data is split between:

- on-chain data
- off-chain data

On-chain data holds parameters such as:

- mint authority
- creator
- royalty splits
- link to off-chain metadata

The Non Fungible Standard from Metaplex.

See [Docs](#)

Off-chain data stores a pair of media file and associated JSON per NFT.

This JSON has anything that describes the individual parameters of this NFT.

▼ Standard		
Field	Type	Description
name	string	Name of the asset.
symbol	string	Symbol of the asset.
description	string	Description of the asset.
image	string	URI pointing to the asset's logo.
animation_url	string	URI pointing to the asset's animation.
external_url	string	URI pointing to an external URL defining the asset — e.g. the game's main site.
attributes	array	Array of attributes defining the characteristics of the asset.
trait_type	string	The type of attribute.
value	string	The value for that attribute.

For example

```
{  
  "name": "SolanaArtProject #1",  
  "description": "Generative art on Solana.",  
  "image": "https://arweave.net/26YdhY_eAzv26YdhY1uu9uiA3nmDZYwP8MwZAultcE?  
ext=jpeg",  
  "animation_url": "https://arweave.net/ZAultcE_eAzv26YdhY1uu9uiA3nmDZYwP8MwuiA3nm?  
ext=glb",  
  "external_url": "https://example.com",  
  "attributes": [  
    {  
      "trait_type": "trait1",  
      "value": "value1"  
    },  
    {  
      "trait_type": "trait2",  
      "value": "value2"  
    }  
,  
  ],  
  "properties": {  
    "files": [  
      {  
        "uri": "https://www.arweave.net/abcd5678?ext=png",  
        "type": "image/png"  
      },  
      {  
        "uri": "https://watch.videodelivery.net/9876jkl",  
        "type": "unknown",  
        "cdn": true  
      },  
      {  
        "uri": "https://www.arweave.net/efgh1234?ext=mp4",  
        "type": "video/mp4"  
      }  
,  
    ],  
    "category": "video",  
  
    // @deprecated  
    // Do not use - may be removed in a future release.  
    // Use on-chain data instead.  
    "collection": {  
      "name": "Solflare X NFT",  
      "family": "Solflare"  
    },  
  
    // @deprecated  
    // Do not use - may be removed in a future release.
```

```
// Use on-chain data instead.  
"creators": [  
    {  
        "address": "xEtQ9Fpv62qdc1GYfpNReMasVTe9YW5bHJwfVKqo72u",  
        "share": 100  
    }  
]  
}  
}
```

Auction

This program supports the running of an auction, currently, only English auctions are supported. It has mechanisms for collecting bids (that can be any SPL token), cancelling bids, stating the winner and withdrawing funds.

The Auction PDA specifies parameters such as start and end time, authority or minimum bid price.

```
#[repr(C)]
#[derive(Clone, BorshSerialize, BorshDeserialize, PartialEq, Debug)]
pub struct AuctionData {
    pub authority: Pubkey,
    pub token_mint: Pubkey,
    pub last_bid: Option<UnixTimestamp>,
    pub ended_at: Option<UnixTimestamp>,
    pub end_auction_gap: Option<UnixTimestamp>,
    pub price_floor: PriceFloor,
    pub bid_state: BidState,
}
```

Token Vault

This program holds tokens being auctioned by the Auction program and allows for fractionalisation of an NFT. Partitioned NFT with a share like structure could be used so that multiple entities can own a single expensive asset.

(NFT) Marketplaces

- Digital assets
- Can be fractionalised
- Can be used as collateral

Magic Eden

The screenshot shows the Magic Eden homepage. On the left, there's a sidebar with links like 'Marketplace', 'Popular collections', 'Drop calendar', 'Auctions', 'Insights', 'Magic Eden List', 'Launchpad', 'Eden Games', 'Creators', 'Intro to NFTs', 'Support', and 'Resources'. Below the sidebar are social media links for TikTok, Discord, Twitter, YouTube, and Instagram. The main content area features a large banner for 'Links Champions' with a cartoon illustration of several golfers. A text overlay says: 'LinksDAO is launching a new profile picture NFT collection called Links Champions - 10,253 unique golf trophies that have come to life! 🎉'. There's also a 'Go to Launchpad' button.

Solsea

The screenshot shows the Solsea homepage. At the top, there's a header with 'ALL-ART UNIVERSE' and a search bar. The main navigation menu includes 'All', 'Art', 'Photography', 'CGI', 'Gaming', 'Utility', 'Collectibles', 'Virtual', and 'More'. On the left, there's a sidebar with sections for 'Create', 'Explore', 'Activity', and 'Stake'. The 'Explore' section includes links for 'NFTs', 'Creators', and 'Collections'. The 'Activity' section includes 'Live Mints', 'Activity', and 'Charts'. The 'Stake' section includes 'Reduce fee'. The central content area features a welcome message: 'WELCOME TO THE OPEN Solana NFT Marketplace'. It describes the platform as the first NFT platform that embeds licenses and unlockable content when minting, with the lowest trading fees on Solana. It encourages users to start trading now with no login required. There are 'Explore' and 'Create' buttons at the bottom. To the right, there's a large image of a virtual room with a woman in a digital space surrounded by floating purple cubes.

Solanart

Trade NFTs at **ZERO** cost

NEW Take out a loan in seconds. [Read more](#)

Buy or sell NFTs and save 100% in fees. The only zero-fee marketplace on Solana.

[Explore collections](#)

[Sell my NFTs](#)



Candy machine

Candy machine has recently been upgraded.

See [Docs](#)

Features

- Accept payments in SOL, NFTs or any Solana token.
- Restrict your launch via start/end dates, mint limits, third party signers, etc.
- Protect your launch against bots via configurable bot taxes and gatekeepers like Captchas.
- Restrict minting to specific NFT/Token holders or to a curated list of wallets.
- Create multiple minting groups with different sets of rules.
- Reveal your NFTs after the launch whilst allowing your users to verify that information.

Candy Machine process flow

1. Create the machine and set up configuration.
2. Add the items we are minting
3. Mint the NFTs
4. Clean up

Candy Guards

There is the ability to add extra features to your candy machine by the use of 'guards'

Example guards

- **Sol Payment:** This guard ensures the minting wallet has to pay a configured amount of SOL to a configured destination wallet.
- **Start Date:** This guard ensures minting can only start after the configured time.
- **Mint Limit:** This guard ensures each wallet cannot mint more than a configured amount.

Sugar

Sugar is a CLI to be used with metaplex, and a replacement for the javascript CLI.

Installation

Mac / Linux / WSL

```
bash <(curl -sSf https://sugar.metaplex.com/install.sh)
```

```
✓ Installation successful: type 'sugar' to start using it.  
gitpod /workspace/SolanaBootcampOctober (main) $ sugar  
sugar-cli 1.1.0  
Command line tool for creating and managing Metaplex Candy Machines.
```

USAGE:

```
sugar [OPTIONS] <SUBCOMMAND>
```

OPTIONS:

-h, --help	Print help information
-l, --log-level <LOG_LEVEL>	Log level: trace, debug, info, warn, error, off
-V, --version	Print version information

SUBCOMMANDS:

bundler	Interact with the bundlr network
collection	Manage the collection on the candy machine
create-config	Interactive process to create the config file
deploy	Deploy cache items into candy machine config on-chain
freeze	Commands for the Candy Machine Freeze feature
hash	Generate hash of cache file for hidden settings
help	Print this message or the help of the given subcommand(s)
launch	Create a candy machine deployment from assets
mint	Mint one NFT from candy machine
reveal	Reveal the NFTs from a hidden settings candy machine
show	Show the on-chain config of an existing candy machine
sign	Sign one or all NFTs from candy machine
thaw	Thaw a NFT or all NFTs in a candy machine
unfreeze-funds	Unlock treasury funds after freeze is turned off or expires
update	Update the candy machine config on-chain
upload	Upload assets to storage and creates the cache config
validate	Validate JSON metadata files
verify	Verify uploaded data
withdraw	Withdraw funds from candy machine account closing it

```
gitpod /workspace/SolanaBootcampOctober (main) $ sugar -V
```

```
sugar-cli 1.1.0
```

```
gitpod /workspace/SolanaBootcampOctober (main) $ >|
```

Sugar will then use these settings by default if you don't specify them as CLI options, allowing commands to be much simpler.

Prepare Metaplex

Install the solana CLI if not already installed

Generate a new keypair

```
solana-keygen new --outfile ~/.config/solana/devnet.json
```

Set the config

```
solana config set --keypair ~/.config/solana/devnet.json
```

```
solana config set --url https://metaplex.devnet.rpcpool.com/
```

Check the config

```
solana config get
```

Sugar Quickstart

1. Create an assets directory

```
mkdir assets
```

2. In this directory add the assets and asset json files. You can find an example in the repo

```
{
  "name": "Studious Crab #1",
  "symbol": "CRAB",
  "description": "The Studious Crabs are smart and productive crabs.",
  "image": "0.png",
  "attributes": [
    {
      "trait_type": "accessory",
      "value": "lamp"
    },
    {
      "trait_type": "chair",
      "value": "red"
    },
    {
      "trait_type": "books",
      "value": "blue"
    }
  ],
  "properties": {
    "files": [
      {
        "uri": "0.png",
        "type": "image/png"
      }
    ]
  }
}
```

From the project directory run

```
sugar create-config
```

This will take you through a number of choices, you can read more about the choices [here](#)
At the end it will save a config file which will look like

```
{
  "price": 1.0,
  "number": 10,
  "gatekeeper": null,
  "creators": [
```

```
{  
    "address": "PanbgcTiZ2PveV96t2FHSffiLHXXjMuhvoabUUKKm8",  
    "share": 100  
},  
{"solTreasuryAccount": "PanbgcTiZ2PveV96t2FHSffiLHXXjMuhvoabUUKKm8",  
"splTokenAccount": null,  
"splToken": null,  
"goLiveDate": "11 Aug 2022 18:19:16 +0000",  
"endSettings": null,  
"whitelistMintSettings": null,  
"hiddenSettings": null,  
"freezeTime": null,  
"uploadMethod": "bundlr",  
"retainAuthority": true,  
"isMutable": true,  
"symbol": "TEST",  
"sellerFeeBasisPoints": 500,  
"awsS3Bucket": null,  
"nftStorageAuthToken": null,  
"shdwStorageAccount": null  
}
```

Once the config is saved you can upload the files

```
sugar upload
```

You can then create and deploy a candy machine with

```
sugar deploy
```

Verify that it has all worked with

```
sugar verify
```

you can get the details of the candy machine with

```
sugar show
```

You will get a URL to show your candy machine

 Metaplex Candy Machine V2
F2r2Bhh1k1AA9M1uKxDE7QqZGeBFgBrTtdTLZYiGhYjk

Mint Price	0.1 SOL	Mint Stats	Items Available	10	Creators	9xRRzKaWkA..AHRf	100%
Live Date	November 20, 2022 8:42 PM a minute ago	Items Redeemed	0 / 10		Items Remaining	10 / 10	
		Royalties	0 %				

Next mints


Studious Crab #1


Studious Crab #2


Studious Crab #3

Minting NFTs

Now that we have our candy machine deployed we can use it to mint NFTs

```
sugar mint
```

will mint an NFT to your wallet.

Creating a UI for your Candy Machine

Clone the directory into your project folder

```
git clone https://github.com/metaplex-foundation/candy-machine-ui
```

Get your Candy Machine ID, either from

```
sugar show
```

or from the cache.json file

Add the ID to the `.env.example` file and save the file as

```
.env
```

From within the candy-machine-ui directory run

```
yarn install && yarn start
```

Candy Machine SDKs

A [Javascript SDK](#) is available, Swift and Kotlin SDKs are being planned.

[Rust crates](#) are also available

Using the javascript SDK

Setup

You need to setup a connection

```
import { Metaplex } from "@metaplex-foundation/js";
import { Connection, clusterApiUrl } from "@solana/web3.js";

const connection = new Connection(clusterApiUrl("mainnet-beta"));
```

You can then create a Metaplex instance

```
const wallet = Keypair.generate();

const metaplex = Metaplex.make(connection)
  .use(keypairIdentity(wallet))
  .use(bundlrStorage());
```

The metaplex instance then gives the nft module

```
const nftClient = metaplex.nfts();
```

language-ts

which has useful methods

- [findByMint](#)
- [findAllByMintList](#)
- [load](#)
- [findAllByOwner](#)
- [findAllByCreator](#)
- [uploadMetadata](#)
- [create](#)
- [update](#)
- [printNewEdition](#)
- [use](#)

See the full API [here](#)

Example for uploading metadata and creating an NFT.

The ownership and authority information would be taken from the user set up earlier.

```
const { uri } = await metaplex
  .nfts()
  .uploadMetadata({
    name: "My off-chain name",
    description: "My off-chain description",
    image: "https://arweave.net/123",
  };

const { nft } = await metaplex
  .nfts()
  .create({
    uri,
    name: 'My on-chain NFT',
    sellerFeeBasisPoints: 250, // 2.5%
  };

```

Other useful programs

Gumdrop

See [Docs](#)

This helps with creating token and NFT airdrops.

Other considerations

Costs

Solana deployment fees for the collection can be calculated using this online [tool](#)

Arweave deployment fees for the collection can be calculated using this [tool](#).

A collection with 999 items and average off-chain data ~3Mb will cost **1.67 SOL** and **0.87 AR**.

The total price tag for the deployment of this collection at today's Sol and AR valuations would be **~\$30**, but during the peak price of crypto same collection would cost in excess of **\$350**.

Fair to say to anyone considering launching their own collection now is a good time.

The screenshot shows two side-by-side calculators. On the left is the 'Candy Machine Fee Calculator' which has a note about costs for candy machine storage and a note about Arweave being the default upload option. It has a 'Collection Size' input set to 999 and a 'Calculate' button. On the right is the 'Storage Fee Calculator' with a note about Arweave upload costs. It has an 'MB' dropdown menu set to '2997' and a 'Calculate' button. To the right of the calculators is a table titled 'Approximate Cost*' showing fees for Data Size (2997 MB), Arweave (0.874895798188 AR), Winston (874895798188 Winston), and USD (~\$12.93095989721864 USD). A note at the bottom states: '*Fees are dynamic. Pricing is determined by t'.

Approximate Cost*	
Data Size	2997 MB
Arweave	0.874895798188 AR
Winston	874895798188 Winston
USD	~\$12.93095989721864 USD

Marketing and brand management

For an NFT launch to be successful it requires a strategy that includes selecting a target audience, brand creation, marketing or potential celebrity endorsement. Even after the launch continuous community management via a social platform such as discord is likely needed to make sure people do not lose interest especially if the creator is to receive royalties from each resale for perpetuity.

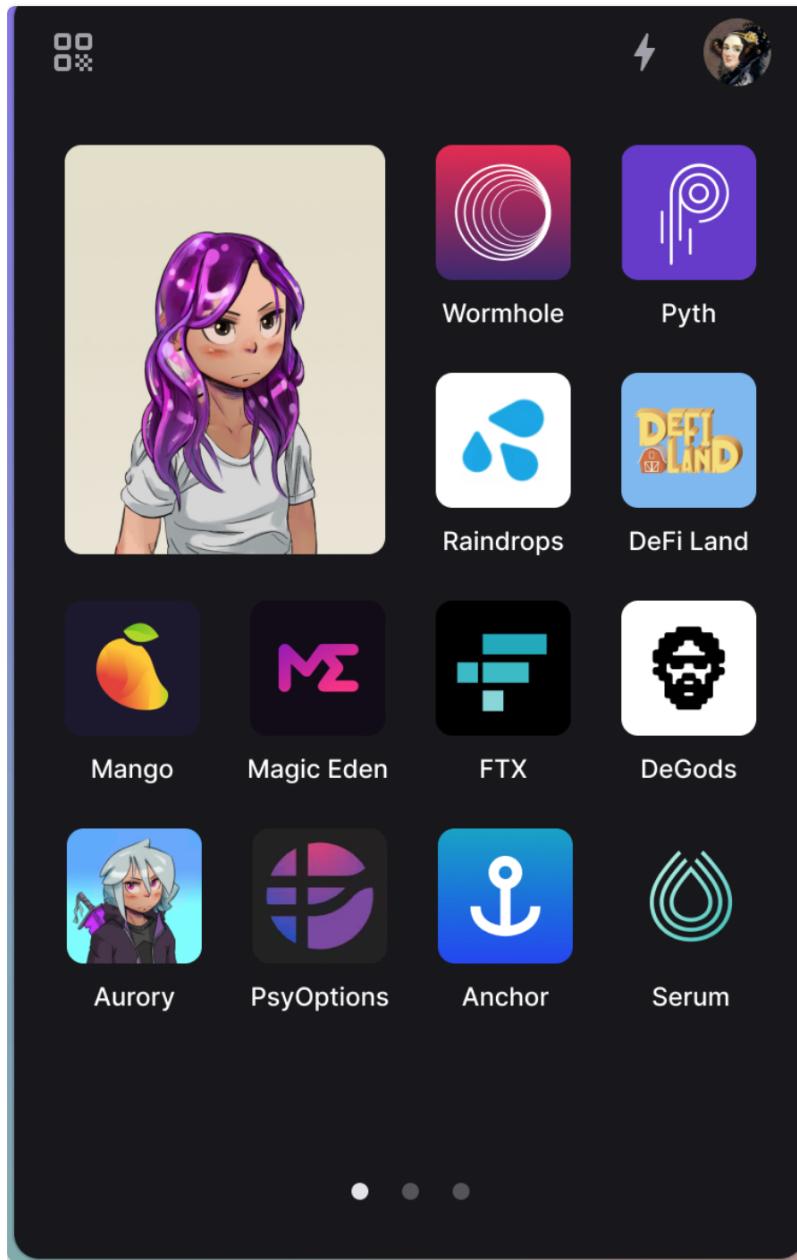
Permanence

It is important to state whether NFT is fixed or whether it can be modified. Generally, NFTs are thought of as immutable and owners might object to suddenly realising their content has been changed. This can be enforced with `is mutable` field being set to `false`.

On the flip side, should NFT be intentionally upgradable, for example, to reflect loalty points or game character attributes it would require `is mutable` field being set to `true`. As Arweave data is immutable, new off-chain data deployment will be required which will incur some fees as well as updating of the Arweave URI.

Either approach is valid as long as it is communicated to the users.

xNFTs



An extension to the NFT concept are xNFTs, the 'x' stands for executable.
"tokenised code, representing ownership rights over its execution"

[Coral](#) have built a multichain wallet [backpack](#) to hold xNFTs in addition to your other assets.

The xNFT standard enables any dapp built on Solana to create an xNFT and be executed natively inside the backpack wallet.

Example use is with Mango Markets, a Backpack user can access their trading account and manage their positions on Mango Market, from the Mango Market xNFT inside the Backpack wallet.

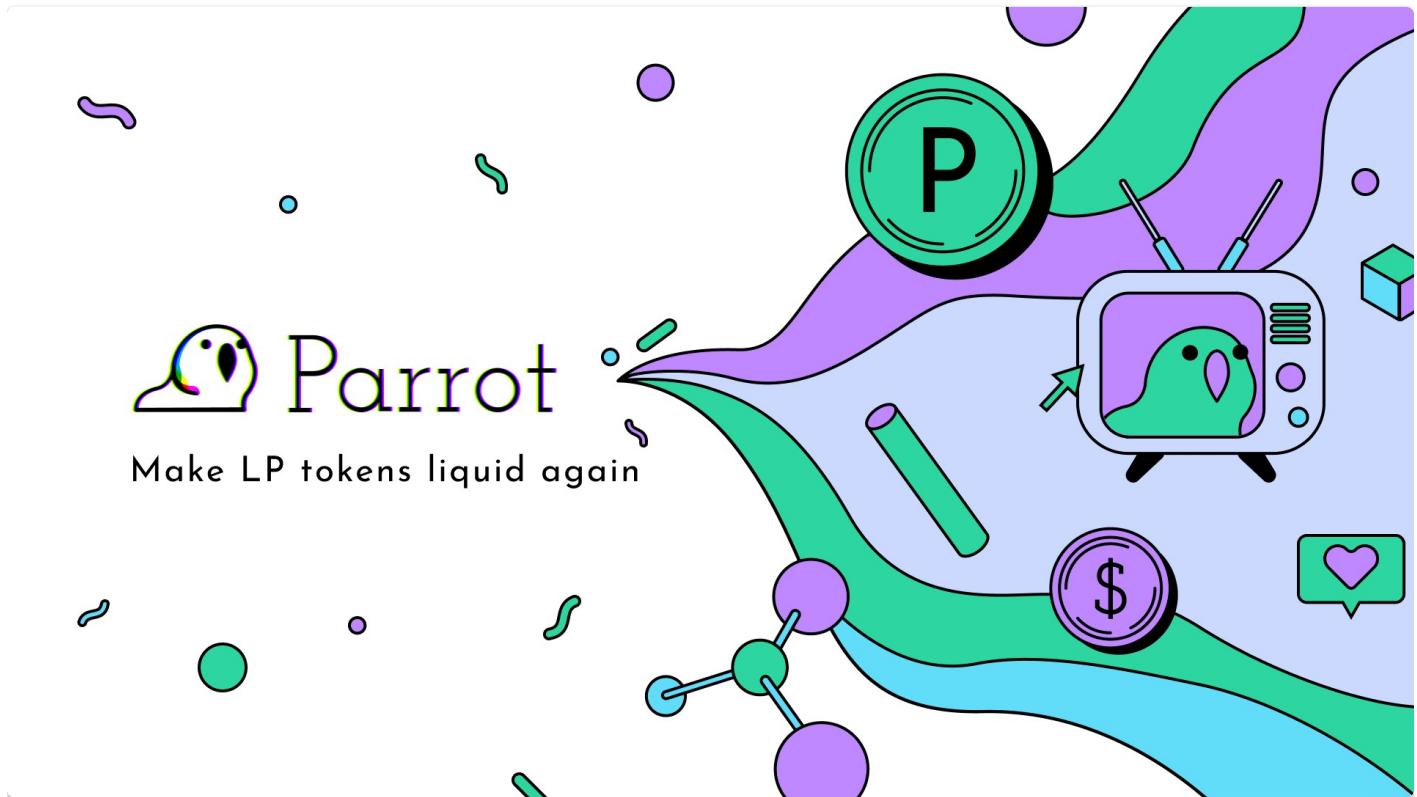
DeFi on Solana

Stablecoins on Solana

From DeFI Llama

Name	Chains	% Off Peg	1m % Off Peg	Price	1d Change	7d Change	1m Change	Market Cap
1 ⓸ USD Coin (USDC)	Ⓝ ⓹ ⓷ ⓸ ⓹ ⓷ +53	0%	+0.12%	\$1.00	-14.83%	-19.28%	-38.24%	\$1.28b
2 ⓒ Tether (USDT)	Ⓝ ⓷ ⓹ ⓷ ⓸ ⓹ +55	-0.12%	-0.20%	\$1.00	-0.06%	-55.57%	-55.85%	\$799.31m
3 Ⓜ Parrot USD (PAI)	Ⓝ	-0.49%	-1.78%	\$1.00	0%	+0.36%	-9.56%	\$17.09m
4 ⓷ UXD Stablecoin (UXD)	Ⓝ	-0.03%	-0.05%	\$1.00	-0.09%	-0.17%	+6.73%	\$16.01m
5 ⓸ Hedge USD (USH)	Ⓝ	-0.01%	-0.88%	\$1.00	+1.98%	+31.50%	-64.14%	\$3,421,554
6 ⓒ USDH (USDH)	Ⓝ	-0.18%	-0.45%	\$1.00	+0.03%	-0.30%	-76.00%	\$2,074,585
7 ⓷ Ratio Stable Coin (USDR)	Ⓝ	-0.78%	-1.22%	\$0.99	-0.06%	-10.33%	-36.81%	\$1,941,241
8 ⓸ Dai (DAI)	Ⓝ ⓹ ⓷ ⓸ ⓹ +34	+0.01%	+0.36%	\$1.00	-0.07%	-7.30%	-14.61%	\$1,402,833
9 ⓸ Binance USD (BUSD)	Ⓝ ⓹ ⓷ ⓸ ⓹ +27	+0.03%	+0.35%	\$1.00	+2.09%	-22.28%	-88.44%	\$472,618
10 ⓸ TerraClassicUSD (UST...)	Ⓝ ⓹ ⓷ ⓸ ⓹ +12	② -97.95%	② -98.06%	\$0.02	-0.36%	-9.72%	-46.86%	\$213,637

Parrot



The Parrot Protocol is a DeFi network built on Solana that will include the stablecoin PAI, a non-custodial lending market, and a margin trading vAMM. These are all use cases designed to solve one single problem: making value locked in DeFi systems accessible.

The Parrot Protocol is setting out to make value locked in LP tokens accessible, by creating a liquidity & lending network collateralised by these LP tokens. T

- Create the PAI stablecoin, backed by LP tokens as collaterals. This creates a common unit of account to make it easy for holders of different types of LP tokens to transact with each other.
 - Create the Parrot Lending market, taking LP tokens as collaterals. This allows LP holders to access their locked value by borrowing against lender liquidity.
 - Create a margin trading product (virtual AMM) using PAI as the common unit of account. This allows the Parrot community to collect fees.
-

Hedge USD

See [Docs](#)

See [White paper](#)



Interest-Free loans

Hedge offers 0% interest vaults, allowing you to indefinitely tap into the value of your assets.



Long-Term leverage

Leveraging with Hedge enables you to keep your leverage positions open for longer for a fraction of the perp funding fees.



Pegged to \$1

USH is always redeemable for its underlying value, ensuring peg.



110% collateral ratio

Hedge offers vaults with collateral ratio as low as 110%, enabling up to 11x leverage.



Collateral backed

USH collateral is kept secure in vaults and is not staked or placed in other protocols.



Efficient liquidations

Protocol solvency is guaranteed by a stability pool that liquidates risky vaults. Anyone can participate to earn HEDGE tokens.

Hedge is a protocol that enables the minting of USH, a stablecoin soft-pegged to the US dollar. USH is minted on flexible terms and gives users instant access to the USH ecosystem.

Hedge allows users to take out 0% interest loans for a one-time fee by depositing collateral for USH. Users are incentivised to keep their collateral-to-debt ratio above 110% to avoid liquidation. When an under collateralized vault is liquidated, users who have deposited USH in a stability pool are returned discounted collateral as a reward. USH is always redeemable for its underlying value, but a fee ensures the protocol is impacted infrequently.

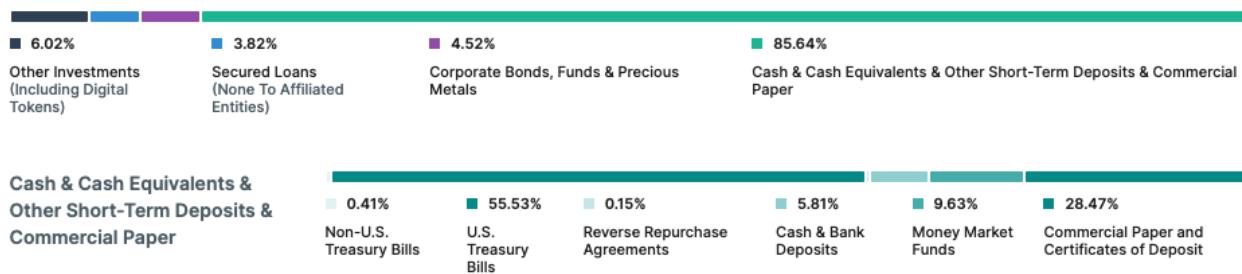
USH Stablecoin

The ceiling price of USH is \$1.10, as users can always deposit the equivalent of \$110 of collateral to mint \$100 worth of USH. If USH is trading above \$1.10, a profit could be made by selling USH on the market.

To establish a floor price for USH, there are two different mechanisms. If a user has a loan and finds USH to be below \$1, they can repay their loan at a lower price by buying more USH, in turn stabilising the price of USH. Alternatively, they can [redeem](#) their USH.

The USH token mint is [9iLH8T7zoWhY7sBmj1WK9ENbWdS1nL8n9wAxaeRitTa6](#).

Reserves Breakdown



UXD Protocol

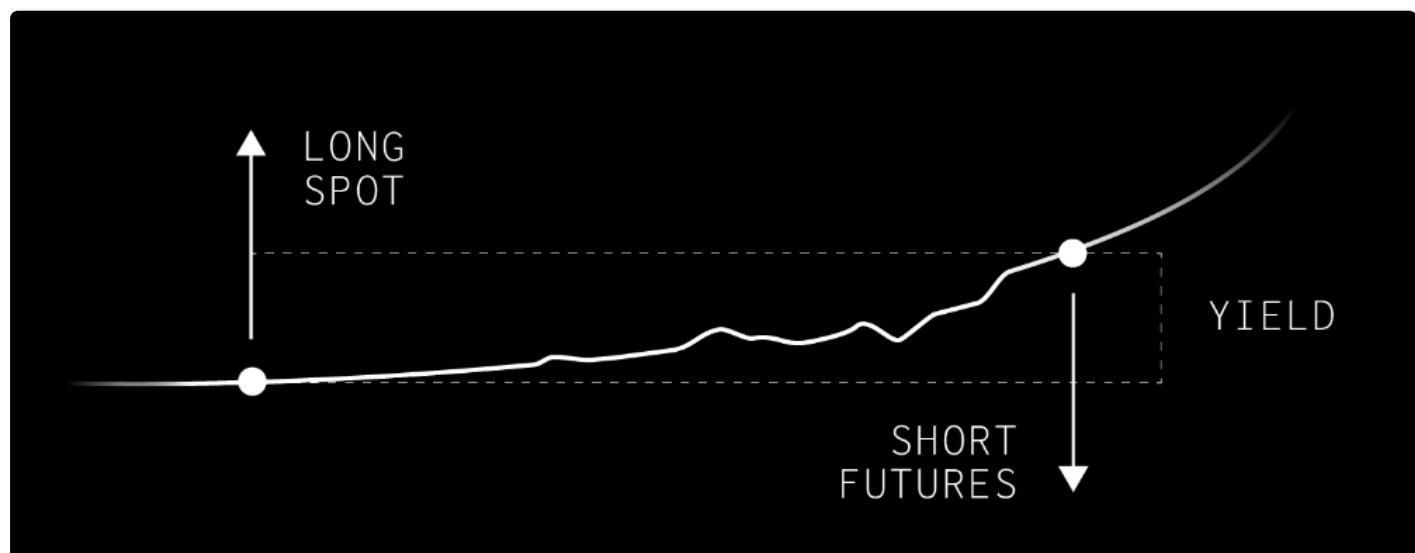
See [UXD](#)

Coindesk [Article](#)

Algorithmic stablecoin backed 100% by a delta neutral position using derivatives.

Launched in January 2022

Aims to have a yield of about 10%



Connected to Mango markets DEX

Serum ecosystem

Serum [whitepaper](#)

Serum is a decentralized exchange program built on Solana where assets can be bought and sold by traders. At the heart of the Serum project is a decentralised order book led by smart contracts, which aims to reflect traditional exchanges by matching buyers and sellers. This gives participants the flexibility to set prices and order sizes when placing orders for the Serum, giving them full control of trading.



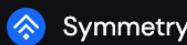
In doing so, Serum is expected to support a new type of decentralised exchange that is expected to compete with those built using Automatic Market Maker (AMM) such as Uniswap, Sushi and Bancor.

Moreover, Serum offers cross-chain support which means investors can trade tokens built on other platforms such as Ethereum or Polkadot. Existing Decentralised Finance (DeFi) projects can also access Serum's features and liquidity, regardless of which blockchain they are built on.

The Serum utility token, SRM, allows holders to receive up to 50 percent discount on trading fees, and participants to vote and participate in the platform's management mechanism.

Those who wish to stay connected about the development of the Serum currency can bookmark their blog for updates.

Serum users



Tag - Index Asset Management

Symmetry is a decentralized index protocol built on Solana blockchain that allows users to easily diversify their portfolios and maximize passive earnings. Users can create an index, include desired crypto-assets along with corresponding weights, and select re-balancing methods.

Links: [Website](#) [Twitter](#) [Github](#)



Tag - Game NFT

Star Atlas is a virtual gaming metaverse emerging from the confluence of state of the art blockchain, real-time graphics, multiplayer video game, and decentralized financial technologies.

Links: [Website](#) [Twitter](#) [Github](#)



Tag - Asset Management

Solrise Finance is a decentralized, trustless, and non-custodial asset management platform built on Solana and Serum. Solrise allows anyone to start or invest in a fund with as little as \$20.

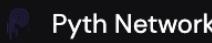
Links: [Website](#) [Twitter](#)



Tag - AMM Trading Yield

Raydium is an AMM and yield farming project built on Solana for the Serum DEX. Unlike other AMMs, Raydium provides on-chain liquidity to a CLOB: funds deposited into Raydium are converted into limit orders which sit on Serum's orderbooks.

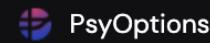
Links: [Website](#) [Twitter](#) [Github](#)



Tag - Oracle

Pyth Network is a specialized oracle solution for latency-sensitive financial data that is typically guarded by centralized institutions. Pyth Network is focused bringing this unique data on-chain and aggregating it securely.

Links: [Website](#) [Twitter](#)



Tag - Options Trading

PsyOptions is an options protocol that aims to bring a mature options infrastructure to DeFi by utilizing the speed and throughput of the Solana blockchain, as well as Serum's CLOB.

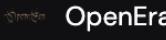
Links: [Website](#) [Twitter](#)



Tag - Borrow-Lending Trading Yield

Oxygen is a DeFi prime brokerage service built on Solana and powered by Serum's on-chain infrastructure. Starting with borrow-lending, users can earn yield and get leverage against their pools, setting the foundation for a vibrant, decentralised financial ecosystem.

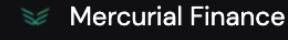
Links: [Website](#) [Twitter](#) [Github](#)



Tag - Game

OpenEra is a PvP, PvE fantasy MMORPG inspired by the likes of RuneScape. OpenEra aims to introduce a unique player driven economy where in-game money markets can be bridged with off-game markets.

Links: [Twitter](#)



Tag - Trading Yield

Mercurial is building DeFi's first dynamic vaults for stable assets on Solana, providing the technical tools for users to easily deposit, swap and mint stable assets. Mercurial will integrate with the Serum orderbook to provide liquidity to the ecosystem.

Links: [Website](#) [Twitter](#)



Tag - NFT Trading

Bridgesplit is the financial infrastructure for cross-chain non-fungible tokens, providing the composability and liquidity needed for the next generation of digital assets.

Links: [Website](#) [Twitter](#)

01 O1 Protocol

Tag - Trading Options

Experience truly autonomous contracts, instant liquidity, millisecond transaction speeds, and DeFi-wide low fees. By traders. For traders.

Links: [Website](#) [Twitter](#)



Tag - AMM Trading

Atrix is the only Serum AMM that allows you to easily create liquidity pools and farms. Get started. Built on Solana. Warp Speed.

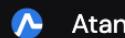
Links: [Website](#) [Twitter](#)



Tag - AMM Trading Borrow-Lending

Leveraged long and short swaps on Solana.

Links: [Website](#) [Twitter](#)



Tag - Asset Management Trading

Atani DEX is the most advanced DEX aggregator on Solana. It powers a trader-friendly terminal, including Technical Analysis, Portfolio Tracking, Price Alerts, and brings Advanced Orders to Serum-based DEXs (Stop Loss, Take Profit, and more).

Links: [Website](#) [Twitter](#)



FRAKT

Tag - NFT Trading

Fraction is a platform for creating liquid markets for NFTs on Solana based on Serum. It includes NFTs fractionalization, Serum GUI and liquidity pools to allow users mint and trade fractions of NFTs.

Links: [Website](#) [Twitter](#)



Tag - Trading

Spin is a premier multichain DEX that enables limited-risk, trustless, and censor-resistant interaction with cryptocurrency assets via futures and options.

Links: [Website](#) [Twitter](#)



Tag - AMM

Cyclos is a concentrated liquidity market maker built on Serum, it brings the familiar constant product market making (CPMM) AMM experience to the Solana DeFi ecosystem.

Links: [Website](#) [Twitter](#)



Tag - Options AMM Trading

Zeta (ζ) is the premier under-collateralized DeFi derivatives platform built on Solana. Zeta Markets leverages the Serum orderbook with an options AMM to provide deep liquidity and a seamless trading experience.

Links: [Website](#) [Twitter](#)

Order Book based DEX

Orderbooks represent a abstract data structure which is manipulated via a matching engine, to match (fill, partial fill) orders

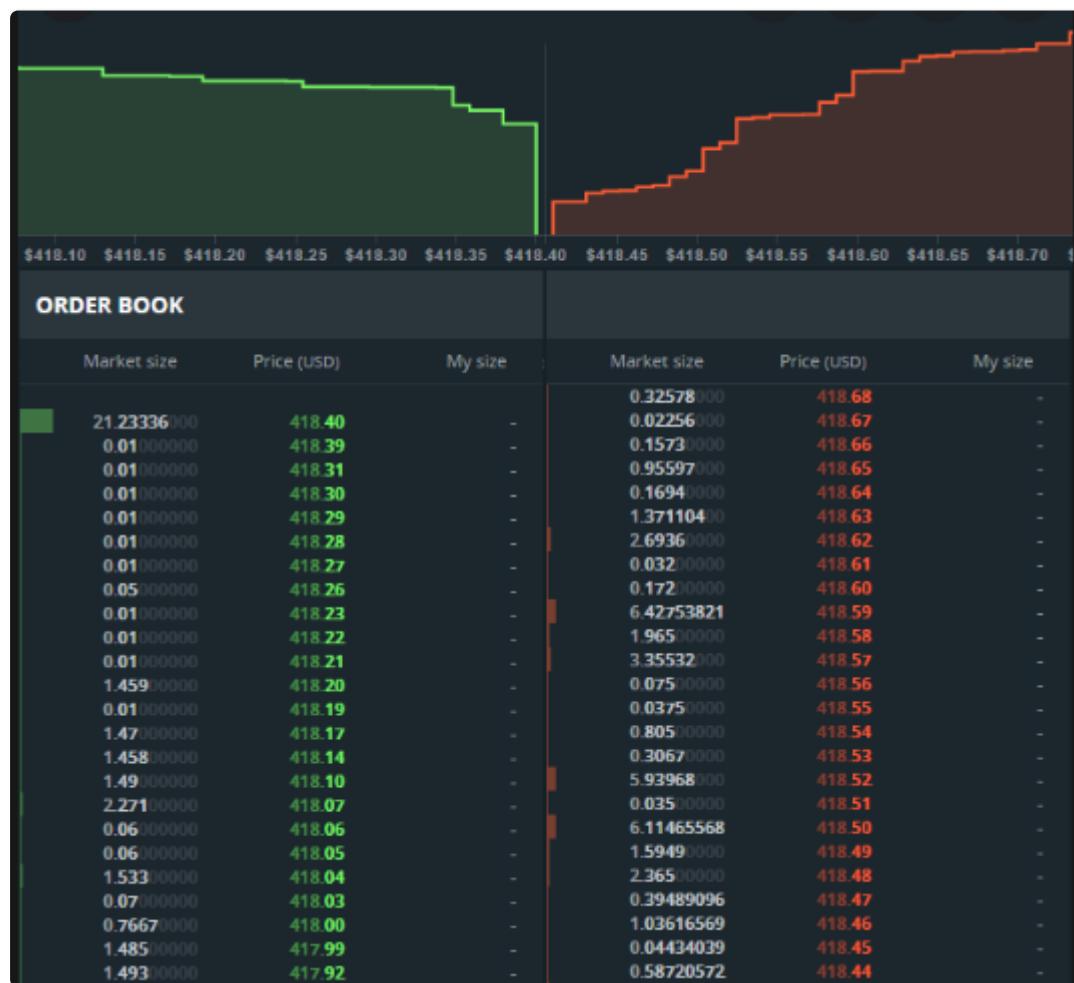
For an indepth explanation of how this is achieved on Serum see this [video](#)

Orderbooks are among the more computationally intensive products in DeFi.

The Serum protocol includes a large number of optimisations to the matching engine in order to make the order matching faster and cheaper.

These orderbooks then serve as the core pricing source for Serum. There is a fee charged on each trade.

All of those net fees go into a buy/burn of SRM. Furthermore, holding SRM decreases your fees paid.



Crosschain swaps

Swaps across chains need to be atomic and frictionless, Serum includes these in its exchange.

Usually accomplished via 'bridges', the implementation of swaps varies and if arbitrators are involved there is the issue of trust.

With Serum, only oracles are needed, rather than a trusted multisig party

Serum allows swaps by providing a contract on Ethereum which can act as an arbitrator and hold collateral for swaps between other chains, for example allowing users to swap between the Ethereum and Bitcoin chains.