



NEAR Data Stack

Easier and Faster with Pagoda

Yijia (Tiffany) Gao

Product Manager

#PagodaisNEAR

Slides:  @pagodaplatform

❖ Hi – I'm Yijia, Tiffany! 🖐️

- Product Manager @ Pagoda
- Computer Science, Data Science, Business, Media @ Columbia University + NYU
- Joined NEAR Family in 2020 and founder of [NEAR Data Guild](#)
- NFT of me! Thank you NEARbits!
- Love traveling, Barca fan, like to mess with my hair color and have a bartending license
- 📈 🦊 🤪 🪄 🎻 🎸



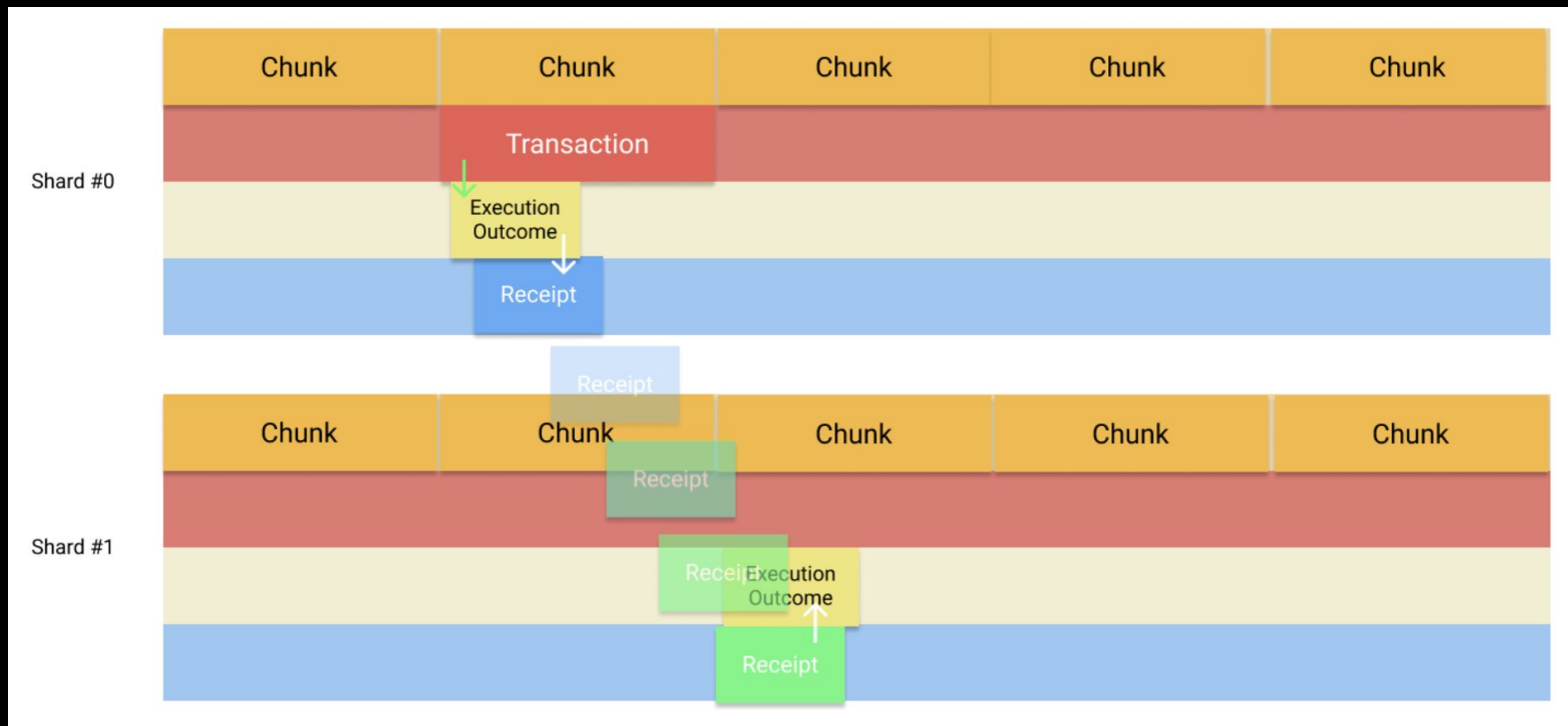


Agenda

- NEAR Lake
 - Intro + NEAR Data Flow + Events
 - Demo listening on Ref txs
 - NFT Tutorial Walkthrough
- Enhanced API - upgraded data query experience
- Pagoda Console - deploy in 15 mins with Alerts, Analytics, RPC Stats
- NEAR Data Stack Next Step



NEAR Data Flow





NEAR Event Standard

Motivation

NEAR and third-party applications need to track mint, transfer, burn events for all NFT-driven apps consistently. This extension addresses that.

Event Log is what enables this standardization

JSON string should have the following interface:

```
// Interface to capture data about an event  
// Arguments  
// * `standard`: name of standard, e.g. nep171  
// * `version`: e.g. 1.0.0  
// * `event`: type of the event, e.g. nft_mint  
// * `data`: associate event data. Strictly typed for each set {standard, version, event} inside corres  
interface EventLogData {  
  standard: string,  
  version: string,  
  event: string,  
  data?: unknown,  
}
```



NEAR Event Standard

```
use near_sdk::log;

// ...
log!(
    r#"EVENT_JSON:{"standard": "nepXXX", "version": "1.0.0", "event": "YYY", "data": {"token_id": "{}"}
    token_id
);
// ...
```

Copy

Valid event logs:

```
EVENT_JSON:{
  "standard": "nepXXX",
  "version": "1.0.0",
  "event": "xyz_is_triggered"
}
```

Copy

```
EVENT_JSON:{
  "standard": "nepXXX",
  "version": "1.0.0",
  "event": "xyz_is_triggered",
  "data": {
    "triggered_by": "foundation.near"
  }
}
```

There is a known
limitation of 16kb
strings when
capturing logs



Beyond Indexer Framework



NEAR Lake: Stream blocks to your server

How does it work?

Define the accounts, contracts, events you want to listen to and get the data streamed to your preferred destination

```
<block_height>/  
block.json  
shard_0.json  
shard_1.json  
...  
shard_N.json
```

`<block_height>` is a 12-character-long `u64` string with leading zeros (e.g "000042839521"). See [this issue](#) for a reasoning.

`block.json` contains JSON-serialized `BlockView` struct. **NB!** this struct might change in the future, we will announce it

`shard_N.json` where N is `u64` starting from `0`. Represents the index number of the shard. In order to find out the expected number of shards in the block you can look in `block.json` at `.header.chunks_included`



NEAR Lake: Stream blocks to your server

Reduce monthly costs by 25x

Spin up time reduced by 5x

Low Maintenance

JavaScript, Python, Rust and more

Get started in 6 minutes





NEAR Lake Demo

Daniel Wang



Get started from Pagoda.co!

Pagoda

NewRpcStatsProject

Mainnet

Pagoda Demo - Dev

<> APIs Beta

Alerts

Contracts

Analytics

Indexers

Settings

Logout

Dark Mode

Create your own indexer using NEAR Lake!

Productive

Build an indexer using our S3 JSON storage. No need to standup and sync a full node.

Resilient

Indexer got stuck? No worries! Restart from any block and run it immediately.

Immediate Access

Connect with the full history of the NEAR blockchain in milliseconds.

Familiar

Write indexers in languages you're already familiar with: JavaScript, Python, or Rust.

NEAR Lake Framework allows you to build your own indexer that subscribes to the stream of blocks from the NEAR Lake data source and create your own logic to process the NEAR Protocol data.

Here are some tutorials to help you get started:

- [near-lake-raw-printer](#) : simple example of a data printer
- [accounts-watcher](#) : indexer example that watches for transactions related to specified accounts/contracts
- [nft-indexer](#) : a working NFT indexer

Try Out NEAR Lake

[Terms of Use](#) [Privacy Policy](#)



Enhanced API: Query token balances

Keys Statistics **Enhanced API** Explorer Soon

Want to interact with RPC API? Check out the key tab for the setup guide. Dismiss

NEAR Enhanced API powered by Pagoda

Overview

ENDPOINTS

NFT

- Get NFT GET
- Get NFT history GET**
- Get user's NFT collection overview GET
- Get user's NFT collection by contract GET
- Get NFT contract metadata GET

Coins

- Get user's coin balances GET
- Get user's NEAR balance GET
- Get user's NEAR history GET
- Get user's coin balances by contract GET
- Get user's coin history by contract GET
- Get FT contract metadata GET

SCHEMAS

- CoinBalancesResponse
- FTContractMetadataResponse
- HistoryResponse
- MetadataResponse
- NearBalanceResponse
- NFTCountsResponse
- NFTResponse
- NftsResponse

Get NFT history

GET [https://near-testnet.api.pagoda.co/NFT/\(contract_account_id\)/\(token_id\)/history](https://near-testnet.api.pagoda.co/NFT/(contract_account_id)/(token_id)/history)

This endpoint returns the transaction history for the given NFT and timestamp / block_height. Note: The result is centered around the history of the specific NFT and will return list of its passing owners and metadata.

Auth

x-api-key : 123

Parameters

contract_account_id : string
token_id* : string
limit : integer

Send API Request

Request Sample: Shell / cURL

```
curl --request GET \
  --url https://near-testnet.api.pagoda.co/NFT/(contract_account_id)/(token_id)/history \
  --header 'Content-Type: application/json' \
  --header 'x-api-key: 123'
```

Response Example

```
{
  "block_height": "string",
  "block_timestamp_nanos": "string",
  "history": [
    {
      "block_height": "string",
      "block_timestamp_nanos": "string",
      "cause": "string",
      "new_account_id": "string",
      "old_account_id": "string",
      "token_id": "string"
    }
  ]
}
```

Limitations

- For now, we only support NFT contracts that implement the Events NEP standard.
- We currently provide the most recent 100 items. Full-featured pagination will be provided in later phases.

Request

Security: API Key

Path Parameters

contract_account_id string required
token_id string required

Query Parameters

limit integer<int32>

Responses

200 500 500

- Indexing over 1k tokens under all NEP standards
- Balances of FTs, NFTs and (MTs soon)
- Balance history
- Metadata of FTs and NFTs
- Interactive Docs in Pagoda!

We want Feedback on what other APIs you want to see next! Feedback tool in Console!

* Support legacy FT and NFT by end of year and more to come in the future!



Build on Pagoda in 15 minutes

Create New Project

Launch & Explore a Project

Want to take console for a spin? Deploy an example project in one-click to get started!

Non-fungible Token (NFT)
Deploy a non-fungible token contract and mint an NFT using contract triggers.

Fungible Token (FT)
Deploy a Fungible Token contract and mint a token using contract triggers.

Guest Book
Sign in with NEAR and add a message to the Guest Book.

Details Interact Contract ABI

oztantest.testnet

2. Function

Select Function
nft_mint

token_id: string

receiver_id: string

token_metadata: JSON

3. Transaction Parameters

Gas: 10 Tgas

Deposit: 0 yocto

Send Transaction

Alerts

2,337,838
Total number of transactions
P: 100% - max 100,000 per week

1,144
Total number of blocks
P: 100% - max 10,000 per week

3,074
Total number of nodes
P: 100% - max 100,000 per week

192
Unlabeled blocks
P: 100% - max 10,000 per week

53,399
Transactions
P: 100% - max 10,000 per week

56
New nodes
P: 100% - max 10,000 per week

64,626
Total number of blocks
P: 100% - max 10,000 per week

Launch

Interact

Grow



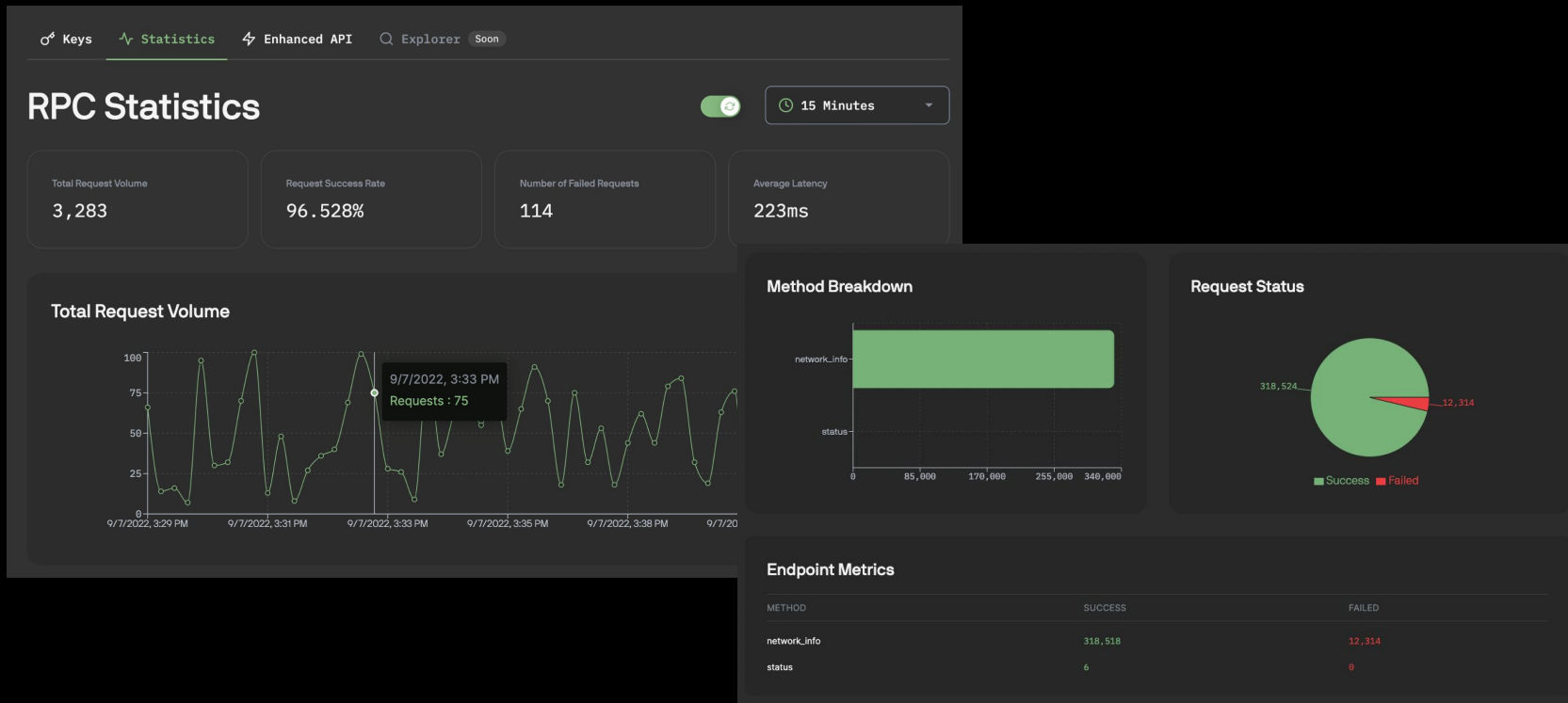
Meet the easiest
Web3 Startup
Platform.

Pagoda.co





RPC Statistics: Get RPC usage insights





NEAR Data Stack Mission

Unlock the full power of data, accelerate your build, operations and growth

Build better UX

*Quick start, less effort on
indexers and integrations*

More Observability

*Visibility into your
infrastructure and contracts*

Business Insights

*Customer insights for your
growth strategy*



NEAR Lake Enhanced API Alerts RPC Statistics Analytics



What's Next?



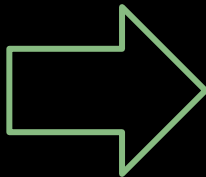
Data infra that just works

RPC Node

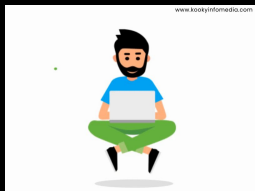
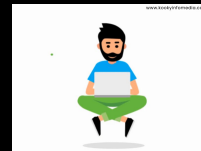
Indexer



Google Cloud



 Pagoda API





Questions?

