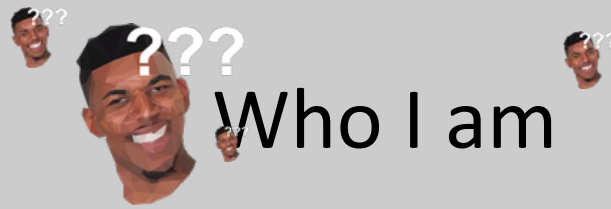


“It is not wrong to go back for that  
which has been forgotten”



**RAVENLAND**  
COMMUNITY RAN COMMUNITY OWNED

Adam Bull CTO, Ravenland  
[adam@ravenland.org](mailto:adam@ravenland.org)  
[www.ravenland.org](http://www.ravenland.org)



- 🦅 Adam Bull aka 'push', Founder of Ravenland.org
- 🦅 Blockchain analyst and automation engineer.
- 🦅 Previously Linux Administrator, Cloud Infrastructure engineer working in Openstack managed cloud.
- 🦅 Managing API + cloud infrastructure fleet of 100,000 hypervisors in 5 Intl. Datacentres.
- 🦅 Using IBM compatible since 82086 AT.

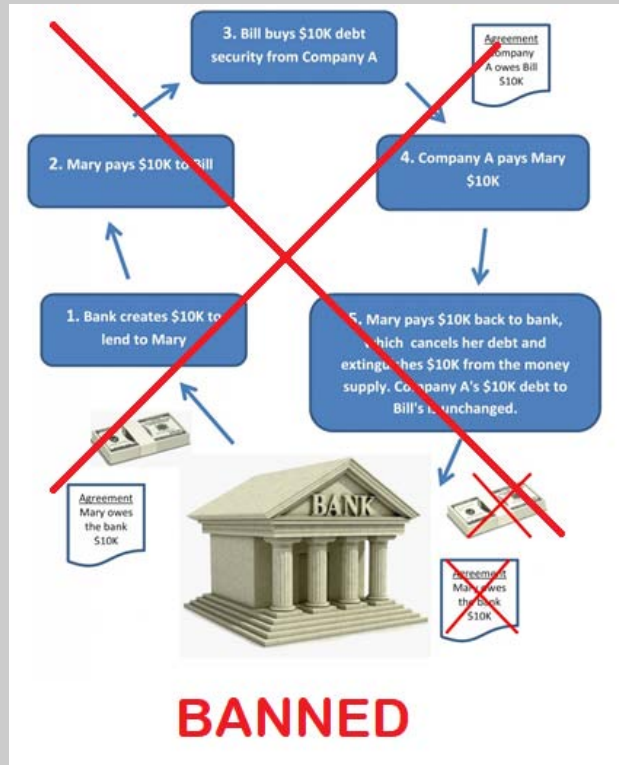
# What is Ravenland?

- 🌀 A Blockchain Software-house.
- 🌀 Small team. consisting of only 1 full time and 1 part time developer 'push' and 'xittz'.
- 🌀 First **community ran and owned PaaS STO** on Ravencoin.
- 🌀 Ravenland token to equal working equity of **Ravenland software** business, **Ravenland Equity Group** and **Ravenland community- estate** group.
- 🌀 We want to create an **open-source market-maker** like Amazon, Ebay and Facebook that the **community can own and control for themselves**.
- 🌀 We want to create an Honest, non-debt based economic system using Ravencoin funded and ran by the community it serves.
- 🌀 A foundation for a fairer economic and governance model to fund businesses in a 'stocks market v2'. 'A Community Wallstreet 2.0'.
- 🌀 Childhood dream was to found my own Island. I created crypto Island Shares. LTD but quickly realised I'd need 'Raven land's software' to achieve the dream.



# GOAL 1

## A Non debt based economy.



## GOAL 2

- Make it open source



RAVENCOIN



*LIGHTNING BITCOIN*

**GitHub**

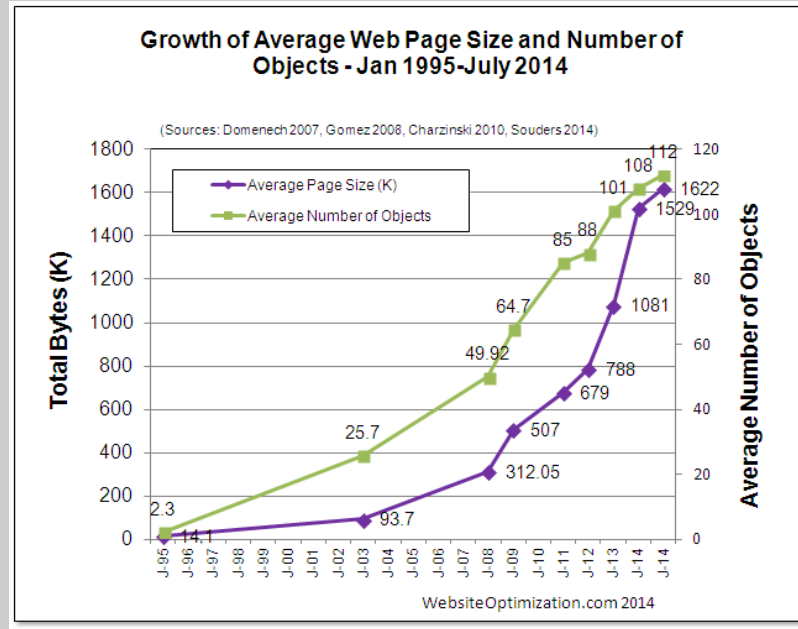
# GOAL 3

- Make it Community Driven










# Why Ravencoin and IPFS?

- Exponential Growth (blockchain CDN)
- Community of DHT peers via HashTable + Secure Ledger Hash.
- Reduce costing
- Guarantee/Trust/Preserve Data
- Store meaning or 'meta' to internet sites and collections of files.
- Useful for maintaining complex or secure Network services.
- The technology that made napster, piratebay and limewire work but with the torrent files put on an immutable ledger blockchain.
- Stop Censorship
- Free Humanity.
- It seems to be needed.
- Security of Bitcoin currency, with ability of storing information about physical trade or physical asset ownership in decentralized or private ways. Ravencoin is "a Bitcoin for the 'real world' of physical assets and trading". Wallstreet 2.0



# What does Ravenland do?

-  We produce Blockchain enabled software
-  Ravencoin Asset Explorer + IPFS Mirror
-  Ravencoin Multi-Asset Faucet
-  Netflix/Plex/Youtube style blockchain video service
-  Kickstarter style blockchain Fundraiser software.
-  RavencoinOS for Raspberry Pi.
-  Provide affordable consultancy services to blockchain startups.  
<https://ravenland.org/consulting>

# What Ravenland software is creating

- 🌀 Ravencoin 2<sup>nd</sup> Layer and Open source blockchain-Retail model.
- 🌀 Ravencoin IPFS Mirror Service
- 🌀 Ravencoin-Asset-Aware API for Enriching Existing and new web-applications or computer games.
- 🌀 Community Software for Loans and Insurance.
- 🌀 Community Estate agency Group + Platform for Ravencoin.
- 🌀 Decentralized and Open Source Land Registration & transfer platforms.
- 🌀 Dividend of company shares thru an STO `security equity` model for Ravenland token holder (*Ravenland Equity Group, Ravenland Commercial Software Group, Ravenland Open Source Community Retail Platform*)

## How do we define Revolutionary software?



- “Revolutionary software, is what people associate with the tech500, like the ‘Googles, and the Twitters’. The Facebooks’ and Instagram’s. but most of that revolutionary software was built using other software or hardware concepts over 5 or 10 years old like Ruby on Rails, PHP And MySQL, Java and Python.
- Revolutionary software isn’t what language you use. It’s what it **lets people do with already-existing ones.**”
- “Revolutionary software often isn’t built on new technology, it’s built on a new *concept* that lets people **do something** they **never thought about doing** before, or something they **never thought they’d be able to do at all.**”
- Ravencoin is built on Bitcoin (Cryptographic currency) and IPFS (Torrents.) Two existing revolutionary technologies. Which is why we close Ravencoin. Because it revolutionizes the use of Bitcoin for Assets and uses Torrent’s DHT technology.

# Ravencoin & It's second layer



Unlike Ethereum ravencoin has no `smart` EVM (Virtual Machine) - 'trusted' ledger or bitcoin's consensus ledger is the key to Ravencoins operation.

Much development logic has been done by Ravencoin team to add asset-token functionality to an 'asset-dumb BTC'.

A second layer still needed to add 'smart functionality' to 'dumb assets' .



## Installing & Running the Ravencoin Linux wallet (reference)

### # Download Ravencoin

```
raventest@terminalserver:~$ wget -q
```

```
https://github.com/RavenProject/Ravencoin/releases/download/v2.2.2/raven-2.2.2.0-x
```

### # Untar/unzip Ravencoin

```
raventest@terminalserver:~$ tar xzf raven-2.2.2.0-x86_64-linux-gnu.tar.gz
```

### # Create a Ravencoin Configuration

```
server:~ $ pico .raven/raven.conf
```

### # Change into Ravend Path

```
server:~ $ cd raven-2.2.2.0/bin
```

```
# Copy Bins to SysENV (optional)
```

```
sudo cp ravend /bin/
```

```
sudo cp raven-cli /bin/
```

### # Start Ravend and RPC listener

```
server:~/raven-2.2.2.0/bin$ ./ravend &
```

```
[1] 22284
```



Check your ravenD process is running with a processID and has bound to port as defined in ~/.raven/raven.conf

```
root@terminalserver:~# netstat -plnt | grep 8766
tcp6      0      0 :::8766          :::*              LISTEN      29926/ravend
root@terminalserver:~# netstat -plnt | grep 8767
tcp       0      0 0.0.0.0:8767      0.0.0.0:*          LISTEN      29926/ravend
tcp6      0      0 :::8767          :::*              LISTEN      29926/ravend
root@terminalserver:~# tail -n 2 /home/raven/.raven/debug.log
2019-06-17 12:46:38 AddToWallet ee99e37897fa08ed88a20d13dc28c62396f09548940c50d48442bac57384e1f6
2019-06-17 12:46:38 Relaying wtx ee99e37897fa08ed88a20d13dc28c62396f09548940c50d48442bac57384e1f6
```

**Ravencoin is now minimally setup for 2<sup>nd</sup> layer software to connect to it.**



When we started we knew nothing about Ravencoin. If you want to learn spend time with raven-cli

[illegible]

raven-cli help is your friend ! It will teach you (nearly) all you will ever need to know about developing with Ravencoin.



# Anatomy of a Ravencoin IPFS\_HASH/ Ravencoins Official Metadata Specification

```
root@terminalserver:~# cat test | jq .
{
  "contract_url": "https://yoursite.com/more-info-about-the-coin.pdf",
  "contract_hash": "<SHA256 hash in hex of contract_url contents>",
  "contract_signature": "<Ravencoin signed contract_hash>",
  "contract_address": "R9x4u22ru3zm5v8suWiXNji4ENWSG7eYkx",
  "symbol": "LEMONADE",
  "name": "Lemonade Gift",
  "issuer": "Lemonade Stands, Inc.",
  "description": "This coin is worth one lemonade.",
  "description_mime": "text/x-markdown; charset=UTF-8",
  "keywords": "Lemonade, Lemonade Stand, Gift, Cold Drink",
  "type": "Points",
  "website_url": "https://lemonadestands.com/redemption_instructions",
  "icon": "<base64 encoded png image at 32x32>",
  "image_url": "https://yoursite.com/coin-image.jpg",
  "contact_name": "Joe Schmoe",
  "contact_email": "joe_schmoe@gmail.com",
  "contact_address": "Lemonade HQ|1234 Nowhere Street|Billings, MT 83982",
  "contact_phone": "207-388-3838",
  "forsale": true,
  "forsale_price": "5000 RVN",
  "domain": "bitactivate.com",
  "restricted": "rule144"
}
```

[https://github.com/RavenProject/Ravencoin/blob/master/assets/asset\\_metadata\\_spec.md](https://github.com/RavenProject/Ravencoin/blob/master/assets/asset_metadata_spec.md)



**RAVENLAND**  
COMMUNITY RUN COMMUNITY OWNED

# Anatomy of a Ravencoin Asset

```
ipfs@ravenland-france ~]$ # List the number of Ravencoin Assets
ipfs@ravenland-france ~]$ raven-cli listassets | wc -l
19639
ipfs@ravenland-france ~]$ # List the assets matching a wildcard
ipfs@ravenland-france ~]$ raven-cli listassets RAVENLAND*
{
  "RAVENLAND",
  "RAVENLAND.ORG",
  "RAVENLANDCASH",
  "RAVENLAND/LOANS",
  "RAVENLANDFAUCET"
}

ipfs@ravenland-france ~]$ # List specific Assets for a service
ipfs@ravenland-france ~]$ raven-cli listassets RAVEN_LAND/FILM*
{
  "RAVEN_LAND/FILMS",
  "RAVEN_LAND/FILMS#1",
  "RAVEN_LAND/FILMS#2",
  "RAVEN_LAND/FILMS#3",
  "RAVEN_LAND/FILMS#4",
  "RAVEN_LAND/FILMS#5",
  "RAVEN_LAND/FILMS#6",
  "RAVEN_LAND/FILMS#7",
  "RAVEN_LAND/FILMS#8",
  "RAVEN_LAND/FILMS#9",
  "RAVEN_LAND/FILMS#10",
  "RAVEN_LAND/FILMS#11",
  "RAVEN_LAND/FILMS#12",
  "RAVEN_LAND/FILMS#13",
  "RAVEN_LAND/FILMS#14",
  "RAVEN_LAND/FILMS#15",
  "RAVEN_LAND/FILMS#16",
  "RAVEN_LAND/FILMS#17",
  "RAVEN_LAND/FILMS#18",
  "RAVEN_LAND/FILMS#19",
}
```

```
[raven@ravenland-france public_html]$ raven-cli getassetdata PICARD
{
  "name": "PICARD",
  "amount": 200000,
  "units": 0,
  "reissuable": 1,
  "has_ipfs": 1,
  "ipfs_hash": "QmZYg9ooTAPpg9HXrn6LfFLKtcfFEfh7bsgNjeyxcX6KjH"
}
```

- How many Ravencoin assets? (COUNT) HOW MANY OTHER RAVENCOIN ASSETS DOES PICARD SIT WITH?
- Which ravencoin assets? (FILTER) WHERE IS THE PICARD ASSET?
- Tell us about the PICARD ASSET.



**RAVENLAND**  
COMMUNITY RUN COMMUNITY OWNED

# Retrieving associated Ravencoin objects from the IPFS network

```
[raven@ravenland-france public_html]$ raven-cli getassetdata PICARD
{
  "name": "PICARD",
  "amount": 200000,
  "units": 0,
  "reissuable": 1,
  "has_ipfs": 1,
  "ipfs_hash": "QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH"
}
```

Get the Details for  
the PICARD Asset

## GET PICARDS' IPFS HASH

```
[raven@ravenland-france public_html]$ curl -I https://gateway.ravenland.org/ipfs/QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH
HTTP/1.1 200 OK
Date: Thu, 13 Jun 2019 21:02:29 GMT
Content-Type: video/mp4
Content-Length: 10674603
Connection: keep-alive
```

Retrieve PICARDS' IPFS File directly via IPFS swarm Network & Then 'Pin' (Save) local to disk

```
root@ravenland-france public_html]# su ipfs
ipfs@ravenland-france public_html]$ cd ~
ipfs@ravenland-france ~]$ ipfs get QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH
saving file(s) to QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH
10.18 MiB / 10.18 MiB [=====]
ipfs@ravenland-france ~]$ ipfs add QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH
added QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH QmZYg9ooTAPpg9HXrn6LfFLKTcfFEFh7bsgNjeyxcX6KjH
10.18 MiB / 10.18 MiB [=====]
ipfs@ravenland-france ~]$
```



**RAVENLAND**  
COMMUNITY RAN COMMUNITY OWNED


# Multi Asset Test Faucet

RAVENLAND

HomeAboutExchangeFilmsMusicAsset ExplorerBlock ExplorerIPFS StatusIPFS Upload

Asset Faucet

Please DO NOT send abusive or offensive tokens to this faucet



Your browsing aFaucet

Capacity 10 Coins

Asset Selector: 0.RVNFREE.COM

Wallet Address: 0.RVNFREE.COM

ASSET\_EXCHANGE/MARKETING

ASSPENIES

BALATERO

BEERMONEY

BRUCELEE

BUYWEEDWITHBTC.COM

CATE

CHADD/KICK\_ROCKS

COINSECPodcast.COM

CORVOCHAT/BETA

CRIPPLE

DIGITAL\_TRENDS/CES2019

DILDO

DUNE/MEDIA

ELLIOTT

ELLIOTT/PART

FORNITE

FRITZERTON

GENIUS.CLUB/2018

Send Asset

Deposits

It is also possible for you to share assets by sending them to `RS7fq8aeJXeDpur4K2o9eX1`

NOTE: There has been some questionable attempts audited due to potential misuse of the service.

This is an experimental service which has been created just for fun for the sole intent of pe understand that you may only do so in the case that your asset is for the use of testing pur can distribute test assets on this faucet.

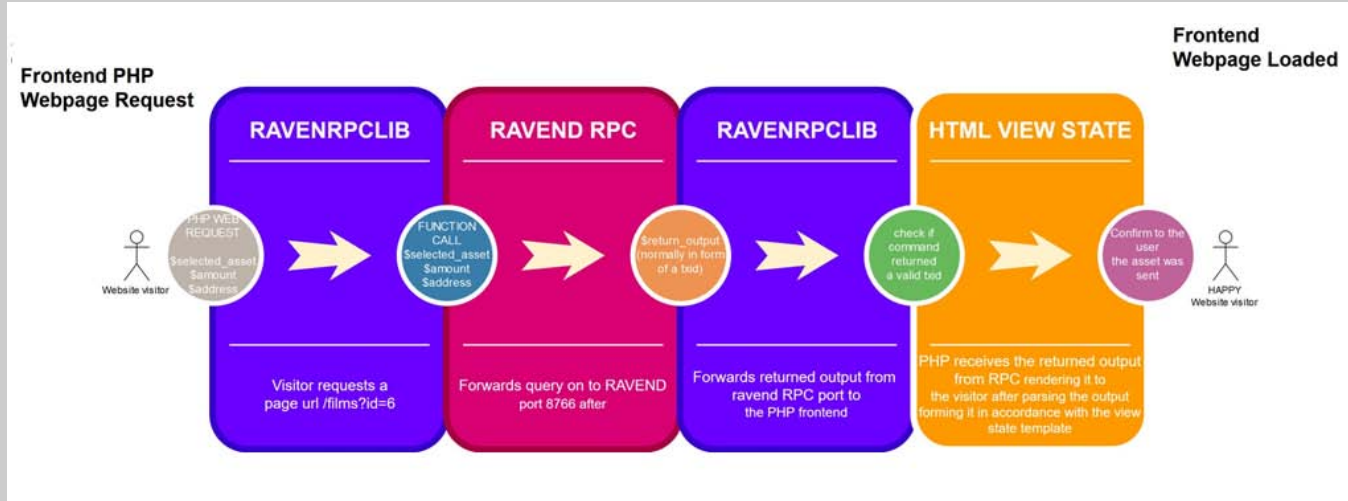
st. All assets sent to the faucet will be

the ravencoin mainnet. By sending tokens to this faucet you nds for some fun. This is not a production exchange however you

# Breaking down the Faucet App

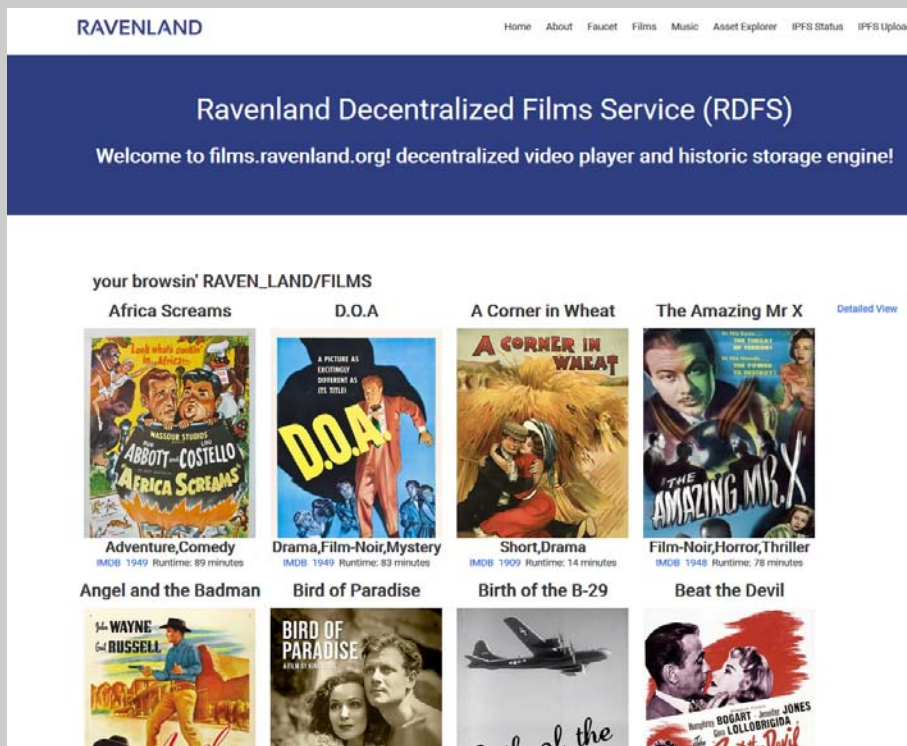
- 🌀 Ravend – **Receives and Sends Assets to and from ravencoin** addresses and checks balances.
- 🌀 Ravencoinrpccli – accesses Ravend's RPC socket via an RPC connector function **'transfer \$assetname \$amount \$address**.
- 🌀 The ravend **RPC socket returns an \$output** to the ravencoinrpccli which is returned to our PHP application.
- 🌀 VIEW STATE: renders the web page you see. And potentially validates that a valid txid was returned confirming the transaction happened successfully.
- 🌀 OPTIONAL: additional validation , checks on number of confirmations  
ravencli gettransaction {txid}. etc

# Breaking down the Faucet Application

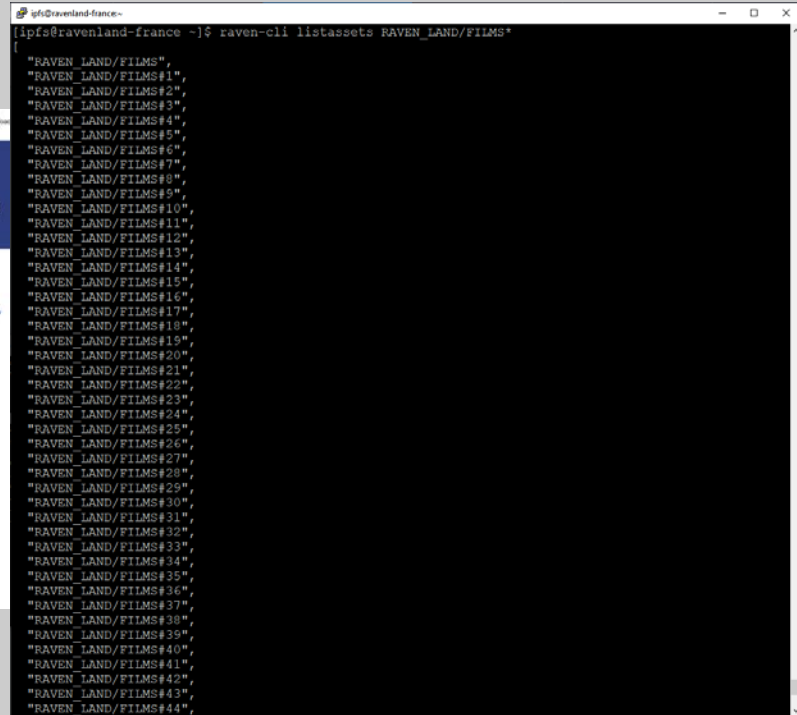
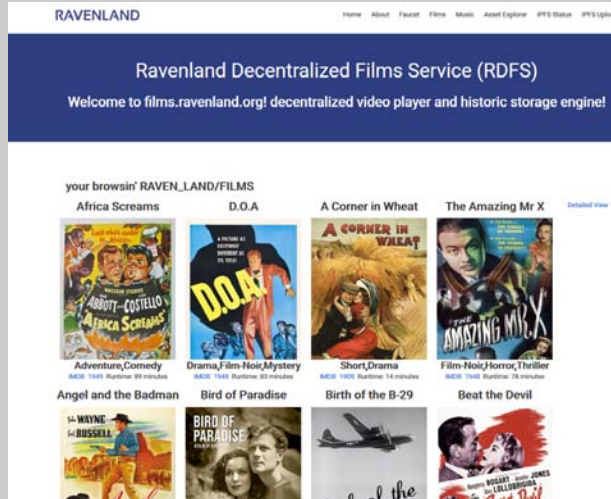


# Decentralized Film and Sharing Service

- USES IPFS
  - USES RAVENCOIN
  - USES RPC CALL
  - RENDERS A VIEW STATE
  - CACHES PHP to HTML
- 
- Examples:
  - `get_assets(RAVEN_LAND)`
  - `get_subassets(RAVEN_LAND/FILMS#*)`
  - `get_filmid($id)`



# How the Data is Represented (List of Films)



# How the Data is Represented (Individual Film id's)

https://films.ravenland.org/getfilm.php?filmid=RAVEN\_LAND/FILMS%2349

RAVENLAND

Home About Exchange Films Music As

## Ravenland Film ID RAVEN\_LAND/FILMS

[Back to Movie Index](#)

Movietitle:	The Last Man on Earth
Genres:	horror,sci-fi
Plot	When a plague devastated life on Earth, the population died or became a sort of zombie living in the dark. Dr. Morgan, the unique healthy survivor on the planet, having a routine life for his own survival: he kills the night creatures along the day and maintains the safety of his house, to be protected along the night. He misses his beloved wife and daughter, consumed by the outbreak, and he fights against his loneliness to maintain mentally sane. When Dr. Morgan finds the contaminated Ruth Collins, he uses his blood to heal her and he becomes the last hope on Earth to help the other contaminated survivors. This new society is scary.
IMDB Link	<a href="http://www.imdb.com/title/tt0058700">http://www.imdb.com/title/tt0058700</a>
Year Published	1964
Runtime	86 minutes



```
[ipfs@ravenland-france ~]$ raven-cli getassetdata RAVEN_LAND/FILMS#49
{
  "name": "RAVEN_LAND/FILMS#49",
  "amount": 1,
  "units": 0,
  "reissuable": 0,
  "has_ipfs": 1,
  "ipfs_hash": "QmeJ7iJjpY9cQt7wUt8epzn24h2DZRpb6e5inCpKHVLhpt"
}
```

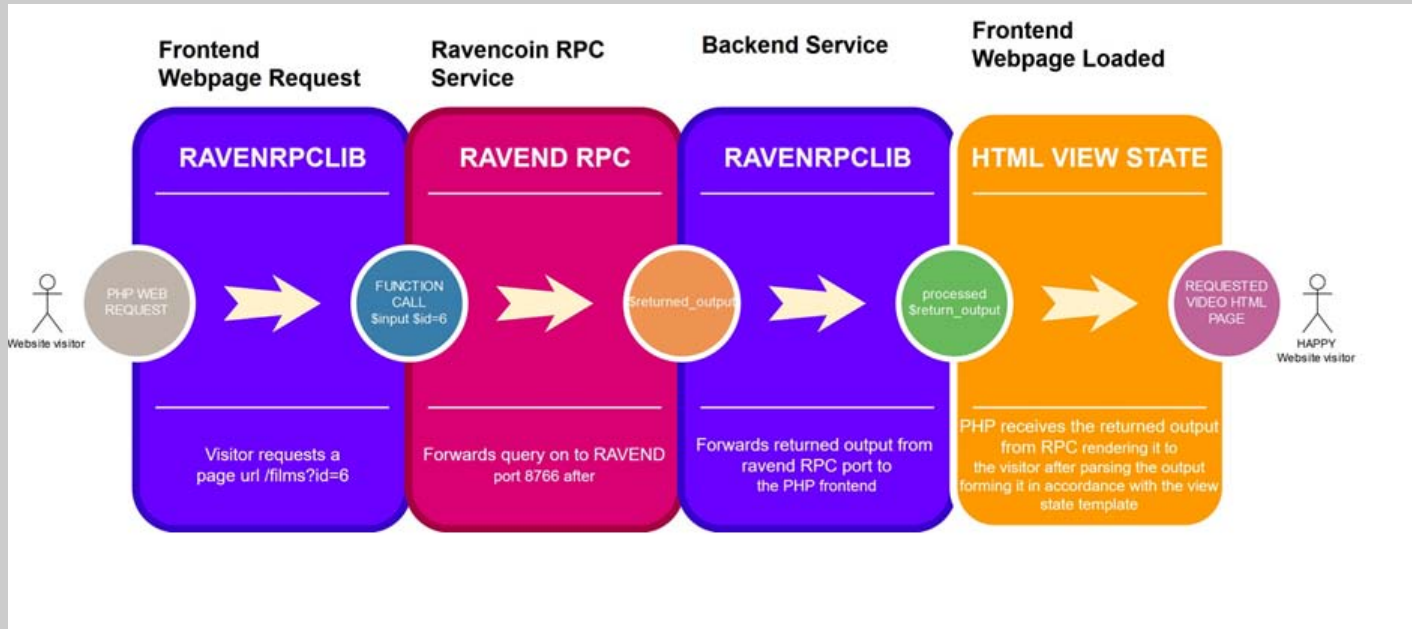
```
[ipfs@ravenland-france ~]$ curl https://gateway.ravenland.org/ipfs/QmeJ7iJjpY9cQt7wUt8epzn24h2DZRpb6e5inCpKHVLhpt
```

```
{
  "contract_address": "RK9AmtmZoiNJGXT5RT0Wp9eNCi2GxlvF9i",
  "symbol": "RAVEN_LAND/FILMS",
  "name": "The Last Man on Earth",
  "issuer": "RAVENLAND.ORG Historic Interplanetary Archive of Culturally Significant Videos",
  "description": "When a plague devastated life on Earth, the population died or became a sort of zombie living in the dark. Dr. Robert Morgan is the unique healthy survivor on the planet, having a routine life for his own survival: he kills the night creatures along the day and maintains the safety of his house, to be protected along the night. He misses his beloved wife and daughter, consumed by the outbreak, and he fights against his loneliness to maintain mentally sane. When Dr. Morgan finds the contaminated Ruth Collins, he uses his blood to heal her and he becomes the last hope on Earth to help the other contaminated survivors. But the order of this new society is scary.",
  "description_mime": "text/x-markdown; charset=UTF-8",
  "imdb_url": "http://www.imdb.com/title/tt0058700",
  "category": "horror,sci-fi",
  "published_year": "1964",
  "length_minutes": "86",
  "type": "Movies",
  "movie_cover_ipfs": "QmRGriZT3TPJs6WkxjfvuotYGTm4NsCBShcndgQ2W6M95J",
  "movie_ipfs": "QmX1lrknAWYNFSDKjKZ6bxcCfJfvxowNx6XP7bx756S9QWP",
  "website_url": "https://films.ravenland.org",
  "contact_name": "Adam",
  "contact_email": "adam@ravenland.org"
}
```

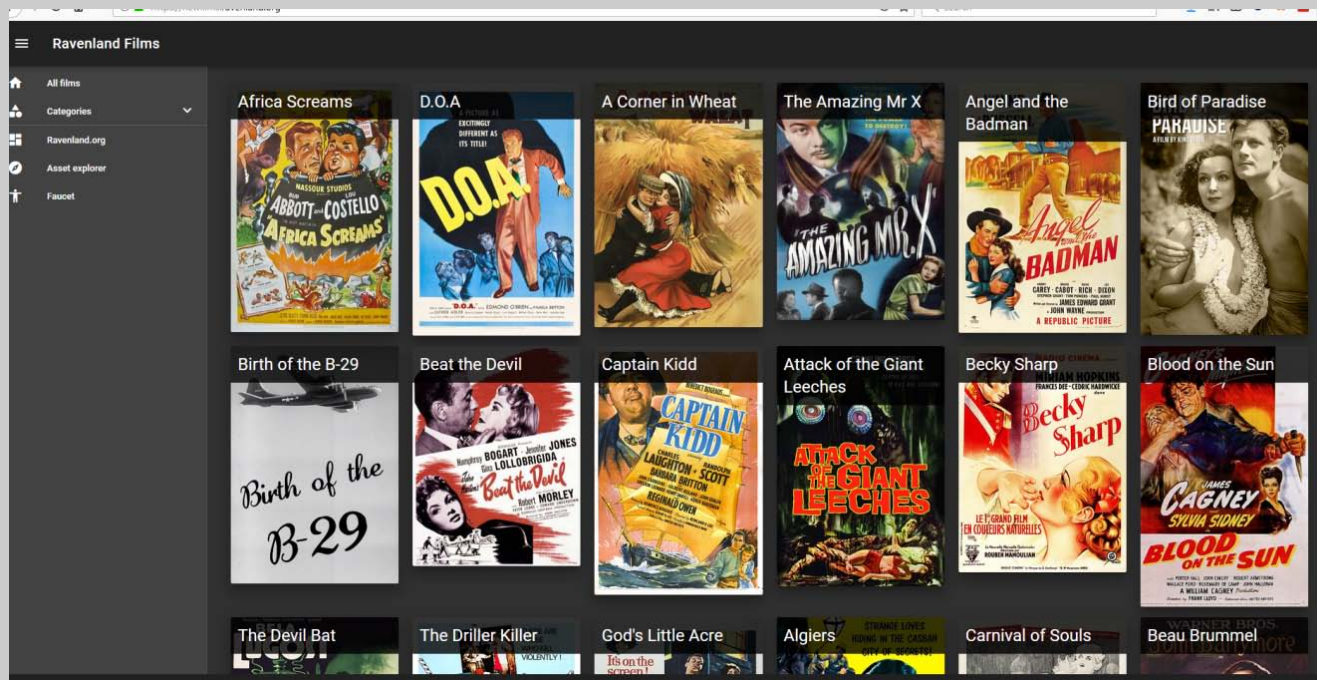
# Breaking down the Films Application

- 🌀 Ravend – **stores and retrieves information** about the Ravencoin blockchain Films, and associated IPFS Objects within it.
- 🌀 Ravencoinrpccli – **accesses the RPC socket of ravend** via an RPC connector function to send a command to ravencoin from the web page query.
- 🌀 The **ravend processes returns an \$output to the ravencoinrpcsocket** which is then returned to our web application (and if necessary filtered accordingly):
- 🌀 Extra Data **retrieved from IPFS via the Ravencoin IPFS\_HASH field.**
- 🌀 After \$output returned to the webpage is **filtered a VIEW STATE** of the object on ravencoin blockchain and IPFS can be rendered.
  - 🌀 i.e. ``

# Breaking down the Films Application





















# Ravenland Films' next version sneak peak



**RAVENLAND**  
COMMUNITY RUN COMMUNITY OWNED

# Breaking down the Blockchain Application In Brief using PHP

```
 $value="RAVEN_LAND/FILMS#33";  
// Retrieve Data from RPC connector  
  
 $unique_asset = $ravencoin->getassetdata("$value");  
  
  
 // Set the title of the film  
 $current_title= $unique_asset['name'];  
  
  
 // Set the Decoding url point from Ravencoin Blockchain for Read-in for IPFS  
 $decodeurl = "https://gateway.ravenland.org/ipfs/" . $unique_asset['ipfs_hash'];  
  
  
 // Retrieve the contents of the IPFS HASH url, decode and populate it into an object orientated code  
 $json = file_get_contents("$decodeurl");  
 $data = json_decode($json,true);  
  
  
 //populate variables in our application with the data from Ravencoin and IPFS.  
 $movietitle = $data['name'];  
 $movedescription = $data['description'];  
 $category = $data['category'];  
 $imdb_url = $data['imdb_url'];  
 $published_year = $data['published_year'];  
 $length_minutes = $data['length_minutes'];  
 $movie_cover_ipfs = $data['movie_cover_ipfs'];  
  
  
 // view state
```



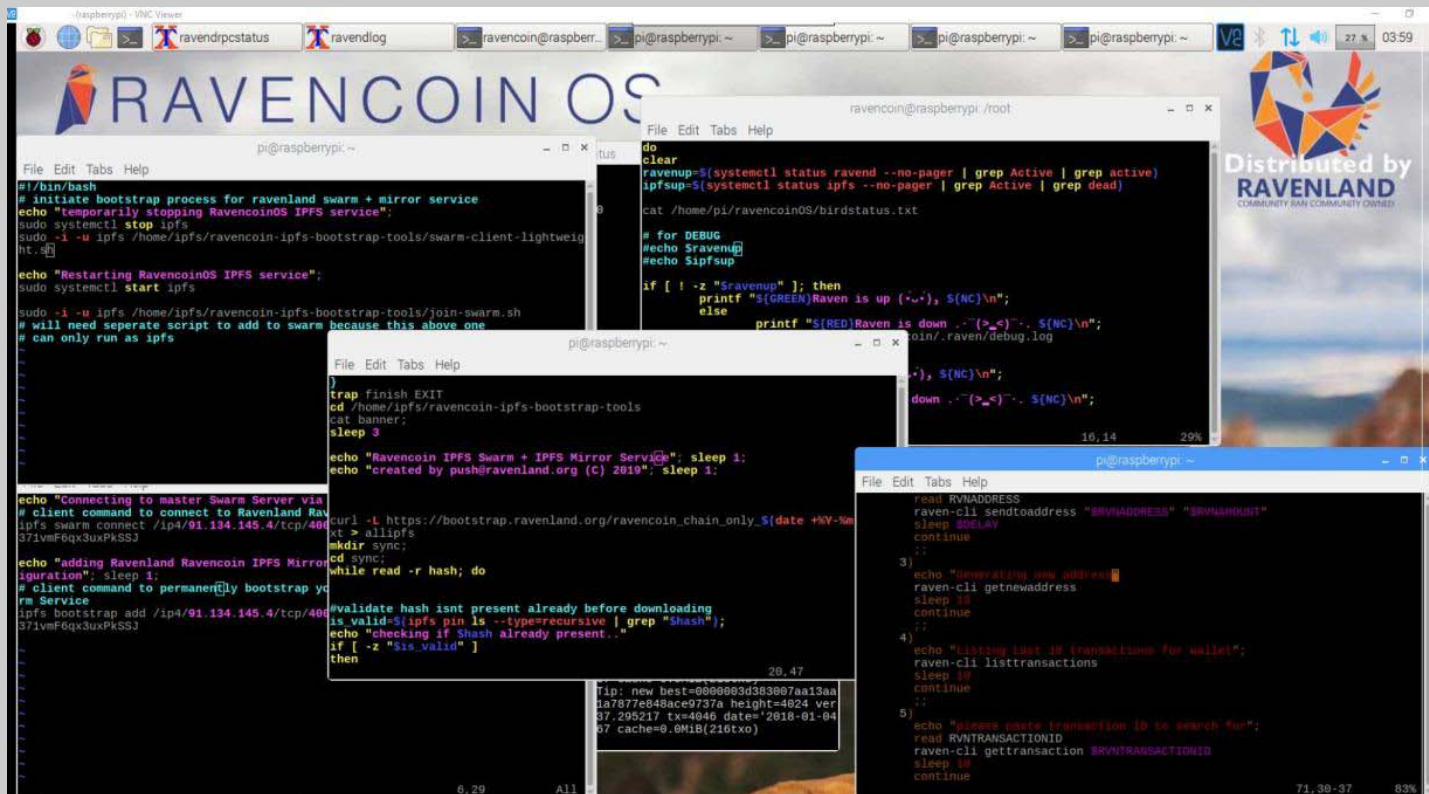
[illegible]

<https://bootstrap.ravenland.org> – Where to download the

<https://wiki.ravenland.org> – RavencoinOSv2 wiki

RAVENLAND  
COMMUNITY RAN COMMUNITY OWNED

# RavencoinOS Helper/Wrapper Scripts (A lot of moving parts to achieve a nearly seamless automation of Ravend and IPFS)



The screenshot displays a Raspberry Pi desktop environment with the RavenCoin OS interface. The background features the RavenCoin logo and the text "RAVENCOIN OS". A sidebar on the right shows the "Distributed by RAVENLAND" logo and the text "COMMUNITY RUN COMMUNITY OWNED". The desktop has several open terminal windows and a file manager window.

The terminal windows show the following scripts:

- Terminal 1 (Left):** A script for managing the RavenCoin OS IPFS service. It includes commands to stop and start the service, and to join a swarm.
- Terminal 2 (Top Right):** A script for managing the RavenCoin OS IPFS service. It includes commands to stop and start the service, and to join a swarm.
- Terminal 3 (Middle Right):** A script for managing the RavenCoin OS IPFS service. It includes commands to stop and start the service, and to join a swarm.
- Terminal 4 (Bottom Left):** A script for managing the RavenCoin OS IPFS service. It includes commands to stop and start the service, and to join a swarm.
- Terminal 5 (Bottom Right):** A script for managing the RavenCoin OS IPFS service. It includes commands to stop and start the service, and to join a swarm.

The file manager window shows the contents of the `/home/pi/ravencoinOS/birdstatus.txt` file, which contains the following text:

```
do
clear
ravenup=$(systemctl status raven --no-pager | grep Active | grep active)
ipfsup=$(systemctl status ipfs --no-pager | grep Active | grep dead)
cat /home/pi/ravencoinOS/birdstatus.txt

# for DEBUG
#echo $ravenup
#echo $ipfsup

if [ ! -z "$ravenup" ]; then
    printf "%s(GREEN)Raven is up (%s), %s(BC)\n";
else
    printf "%s(RED)Raven is down (%s), %s(BC)\n";
fi

if [ ! -z "$ipfsup" ]; then
    printf "%s(GREEN)IPFS is up (%s), %s(BC)\n";
else
    printf "%s(RED)IPFS is down (%s), %s(BC)\n";
fi
```

# How Bootstrap, Assetexplorer + RavencoinOS work together

- Ravenland Scraper – retrieves via RPC ravend assets.
- Ravenland Scraper” Commits ravend assets to a local database schema.
- Ravenland Bridge machine rsyncs the ravend assets database to ravenland france datacentre, and ipfs\_hashes files it had **downloaded in lon1 DC**.
- Ravenland France prod machine creates a list of the synced files pushed to it, generating a ‘small and big’ list at our ‘**publicnet**’ [‘https://bootstrap.ravenland.org/ravencoinOS/](https://bootstrap.ravenland.org/ravencoinOS/)
- RavencoinOSv2 **retrieves the ‘ipfs pin list’ from the Ravenland publicnet france**, and begins mirroring files via the public bootstrap via **ipfs add {}** the remaining files. (ipfs add -r for recursive files or website folders.)
- If RavencoinOSv2 user disables files >100MB it can **greatly save sdcard space**, filtering large files from the OS (reduction from 20G to ~1.8GB).
- If RavencoinOSv2 enables the private swarm toggle, it connects itself to the Private RavencoinOS Swarm Demo Network. (please don’t everyone toggle this! Only people with the swarm.key can access it!! Goal is to provide data to the public network, and build privacy networks too.)

# Querying the Ravenland Asset Explorer Database

- Select all Ravencoin Assets and give us the number of them without an ipfs\_hash
- Show the number of assets which have an ipfs\_hash
- Show the number of unique IPFS hashes in use on Ravencoin blockchain.
- What the database schema looks like for Ravenlands Ravencoin Asset Explorer

```
root@amsterdam: ~  
MariaDB [assetexplorer]> select COUNT(*) from assets where ipfs_hash = ''  
+-----+  
| COUNT(*) |  
+-----+  
|      16136 |  
+-----+  
1 row in set (0.00 sec)  
  
MariaDB [assetexplorer]> select COUNT(*) from assets where ipfs_hash != ''  
+-----+  
| COUNT(*) |  
+-----+  
|      3114 |  
+-----+  
1 row in set (0.00 sec)  
  
MariaDB [assetexplorer]> select COUNT(distinct ipfs_hash) from assets;  
+-----+  
| COUNT(distinct ipfs_hash) |  
+-----+  
|             1089 |  
+-----+  
1 row in set (0.00 sec)  
  
MariaDB [assetexplorer]> select COUNT(distinct mime_type) from assets;  
ERROR 1054 (42S22): Unknown column 'mime_type' in 'field list'  
MariaDB [assetexplorer]> describe assets;  
+-----+  
| Field      | Type          | Null | Key | Default | Extra      |  
+-----+  
| id         | int(12)       | NO   | PRI | NULL    | auto_increment |  
| name       | varchar(32)   | NO   |     | NULL    |              |  
| amount     | bigint(16)    | NO   |     | NULL    |              |  
| units      | int(12)       | NO   |     | NULL    |              |  
| has_ipfs   | int(12)       | NO   |     | NULL    |              |  
| ipfs_hash  | varchar(50)   | NO   |     | NULL    |              |  
| reissuable | int(11)       | NO   |     | NULL    |              |  
| ipfs_mime  | varchar(127)  | NO   |     | NULL    |              |  
+-----+  
8 rows in set (0.00 sec)
```

# A SQL query for all IPFS Movies objects on the Ravencoin blockchain

```
root@amsterdam: ~  
18082 | GREAT ODINS RAVEN | 2000 | 0 | 1 | QmalccryxkfMFIe8LxUezj3E5pac1e1CFy5F4mzuxbu4F2 | 1 | video/mp4 |  
18114 | MLECRYPTOBASEBALL | 1 | 0 | 1 | QmezzVAAJ4HWshwF9Y8L1C8Ycm7WFzqiiGadV9jmQawe8 | 1 | video/mp4 |  
18286 | PICARD/KLINGONSONG | 1000000 | 0 | 1 | QmTfuipU3qce91GzpEw6U6bHrbMKnVihGo26Ty23hnmD7 | 1 | video/mp4 |  
18332 | THE BORING COMPANY | 1 | 0 | 1 | Qmf8n8VMyAT2bGw5Xku2nVn64qF6v9FafPX4msC9bkdkQ45 | 1 | video/mp4 |  
18421 | HISTORIC/1949#D.O.A | 1 | 0 | 1 | QmUzjPySP52FNSknYKXJqHXsqjBYFPFECV1ATKaV7thCPwm | 0 | video/mp4 |  
18443 | MY CHEMICAL ROMANCE | 1 | 0 | 1 | QmaYwKQc8h9GepGHoJ2LvTYtKZe7HHUJNdYoB9VLI9AKA | 1 | video/mp4 |  
18610 | LEGENDS/CARRIEFISHER | 1000001 | 0 | 1 | QmWz59KhdpbQobnwvR55WkjThK8scpKZgPstPaSNHJ2D4 | 1 | video/mp4 |  
18611 | LEGENDS/CHRIS FARLEY | 1000000 | 0 | 1 | QmXfcm3JYbASWH99fVpLejYvyV6oGGAAntyVfvsXb5aBS | 1 | video/mp4 |  
18672 | WILLIAMS#LEAVEATRACE | 1 | 0 | 1 | QmNqMKXenzfbbGwcjGjXCUShykKxGurkN7mDmum21Kpn7a5 | 0 | video/mp4 |  
18727 | JEAN CLAUDE VAN DAMME | 21000000000 | 0 | 1 | QmU4y6MfCeHjUcqReB5uKovBj7rQo4g4DLF1XTDRES7XGwo | 1 | video/mp4 |  
18736 | MIKE/VID OF THE MONTH | 200 | 0 | 1 | QmakpbKWBUnCrW5PFc5Z4ZrXjhqt2LDH8RjMa2Y9fyf8MK | 1 | video/mp4 |  
18748 | RICK AND MORTY/502E01 | 9999999999 | 8 | 1 | QmVgs5eSiFzg4qk5d98auWQGr1z5Uen9ZbBAdMkhLCQAgC | 1 | video/mp4 |  
18822 | DIGITAL_TRENDS/CES2019 | 1800000 | 0 | 1 | QmVjXzYTBod8f8tNJ3jjpRh837Y68FDW1w49N8eEKZbADB | 1 | video/mp4 |  
18867 | SEAL CLUBBER#YourName | 1 | 0 | 1 | QmdGBUDGJU9RrQLF2D1mcDF5J71BCC9H9HEF6V5RDBS5u | 0 | video/mp4 |  
18897 | YOUTUBE/USER/DEMO/MV01 | 50000000 | 0 | 1 | QmWfNer87gEzYp8psxbNYKzCeTV3L4qq2byLppsqvY2pvk | 1 | video/mp4 |  
18947 | IPFS.LIB/MONEY REVEALED | 1 | 0 | 1 | QmWb8Kdmzn47nPFU19k29TiRnq7ukcira2RFdKMH7WHYeq | 1 | video/mp4 |  
18976 | TRAILER/WBP/DTV.PIKACHU | 1000000 | 0 | 1 | QmWqtHy8LEjAW1wmgqP8MSiMXQ46GuxfPkXjVsXz8YqVGD | 1 | video/mp4 |  
19030 | THE JOB.ROGAN.EXPERIENCE | 1 | 0 | 1 | QmVQpQhKngMjF8SVVMdkE2LqB139Y82f8LhVpPYjRgr8xs7 | 1 | video/mp4 |  
19031 | THE BORING COMPANY/MEDIA | 1 | 0 | 1 | QmPKmJlSVQvsmEgrNrhLd1Rf3NjNLNHUjJgr9HfRszsFNUe | 1 | video/mp4 |  
19084 | LINDSEY.STRLING/CHRISTMAS | 1000000000 | 0 | 1 | QmC5M28FcyJ7XihfKAMqftUqVbWgwrENgKXfMexYEMurc | 1 | video/mp4 |  
19091 | PICARD/KLINGONSONG#KAHLESS | 1 | 0 | 1 | QmeZi5g5MBsmoDVe8tptUM6fpfnKEXlUGrtg5ToPmX67Kb | 0 | video/mp4 |  
19095 | SUANLI.IO#2018_ITEM REPORT | 1 | 0 | 1 | QmfdzKjzU5EzffagciZRKeSPM3sAsfpwjeFtsHo1PQVZpJ | 0 | video/mp4 |  
19113 | DEEPPAKES/CELEB#ScarJo1.XXX | 1 | 0 | 1 | QmdwdqW7szsQUSruXVhKKhv317Chmry6QK1R7rdjZL1hd7 | 0 | video/mp4 |  
19119 | IPFS.LIB/MONEY REVEALED#ep7 | 1 | 0 | 1 | QmXluwrDuszBzBoFqQxWhkKXRYjEavdYm4D4nYa5UGW5mbeB | 0 | video/mp4 |  
19120 | MEAN.GIRLS/THAT IS SO FETCH | 1000000 | 0 | 1 | QmFPD6mxsdW19WVUAUDJg9NG3maQyAqsStdtdmRWVfPo881 | 1 | video/mp4 |  
19143 | HISTORIC/1949#AFRICA SCREAMS | 1 | 0 | 1 | QmYat2ARRPgaXsK8FGNkYgXKpYADEt4tDCmgjobrjJgwpJ | 0 | video/mp4 |  
19169 | CRYPTO.BREAKFAST#AvocadoToast | 1 | 0 | 1 | QmSAjPaLSnmf417DHFefXbEuhkALKfd7BCVWzPEhbckqg | 0 | video/mp4 |  
19216 | DEEPPAKES/CELEB#ZoeyDeschanel | 1 | 0 | 1 | QmWnTpGcQHE9pgBbcQL2naXWHfKd3zaqZTVJzUn4ambhi | 0 | video/mp4 |  
19220 | HISTORIC/1949#MALICE IN PALACE | 1 | 0 | 1 | QmYKc4pdyag1X7yTvfFfoUytkr4jdan7fKVPNyXoKvGBWtZ | 0 | video/mp4 |  
19221 | HISTORIC/1949#THE INSPECTOR GE | 1 | 0 | 1 | QmSdJN23piLakFESgdWjYfYfg8fzBgrFRekCm53PyxEqn | 0 | video/mp4 |  
19224 | IPFS.MOVIES#Mr.Peabody&Sherman | 1 | 0 | 1 | QmaF9LlZQrkbkSum5FEnEBLYEmS1lxBY3g42tCXqQyMgy | 0 | video/mp4 |  
19226 | IPFS.MOVIES/JAYSON#KnightsTale | 1 | 0 | 1 | QmY2CWNpNwGwqYhKWCQvcp7NFUgP1ApEqSDRRbnzKSNFZ | 0 | video/mp4 |  
19228 | MACHINEANDMOTION.IO/VIDEO#HAM | 1 | 0 | 1 | QmF5PCDz2XEPu4QA93gSLMgfFwDmLJB1AWvs3FwyYXow6c | 0 | video/mp4 |  
19239 | SUANLI.IO#2018_HASH REPORT_001 | 1 | 0 | 1 | QmfPKrexjY95jzNEooBadMSSsqgMRCoYcwanTyXsXY5Mq2AD | 0 | video/mp4 |  
19240 | SUANLI.IO#2018_HASH REPORT_002 | 1 | 0 | 1 | QmZ86ym77qJABaWKhrhGTZ4KtQPCprqXhpTbbu3KqByq | 0 | video/mp4 |  
19241 | SUANLI.IO#2018_HASH REPORT_003 | 1 | 0 | 1 | QmXfDA6YgQ56GFG5crt7ZmNpYq37VhFRNCJMKdmA4qe7S | 0 | video/mp4 |  
19242 | SUANLI.IO#2018_HASH REPORT_004 | 1 | 0 | 1 | QmZQExqlvTeA9pSkj2neszGEQ1TuxU72PuGE5GJksWQZ6P | 0 | video/mp4 |  
19243 | SUANLI.IO#2018_HASH REPORT_005 | 1 | 0 | 1 | QmQAbFPvAv3cd22XluamQz84pbv7ZvNcqlVKJYwad7Prt | 0 | video/mp4 |  
19244 | SUANLI.IO#2018_HASH REPORT_006 | 1 | 0 | 1 | QmXahPLs6vglYpoKwWdZVvHAKC1kgGEAa8v88DhYKvJXp | 0 | video/mp4 |  
-----  
72 rows in set (0.01 sec)  
MariaDB [assetexplorer]> select * from assets where ipfs mime like 'video/mp4';
```

# Automating Ravencoin Data? How can we do this part 1?

The answer is not that easily. Getting the data is easy. Doing it well is difficult.

Enter some automation.

To get a really good pipeline going it is best to have

1. Dedicated scraper system.
2. Dedicated DB storage system (backend db)
3. Dedicated frontend (consumer of backend db)

```
8b,dPPYba, ,adPPYba, 8b      d8      ,adPPYba, 8b,dPPYba,
88p'  "ys  "      ys  d8'  a8P      88 88p'  "8a
88      ,adPPPP88  `8b  d8'  8pP'      88 88  LAND
88      ,88      `8b,d8'  "8b,  ,aa 88 88  ASSET EXPLORER
88      "8b8dP"ys  "8"      "Yb8d8"  88 88  IPFS SCRAPER

EOF

mysql -u root -D assetexplorer -e "truncate asset_ipfs;"
mysql -u root -D assetexplorer -e "truncate assets;"
mysql -u root -D assetexplorer -e "truncate asset_json;"
mysql -u root -D assetexplorer -e "truncate assetaddresses;"

cd /mnt/[redacted] database2

# old path
# cd /mnt/[redacted] database2

# run db batch run
./makedatabase.sh

# give grace time, not necessary but seems polite
sleep 10

# dump database
cd automation
#clean mess
rm assetexplorer-latest.sql

mysqldump -u root assetexplorer > assetexplorer-latest.sql

# copy database to remote prod
scp -rp assetexplorer-latest.sql [redacted] /root/assetexplorer-latest.sql

# install database to remote prod

# last run
# for sanity if this fails frontend will die
ssh [redacted] 'mysql -u root assetexplorer < /root/assetexplorer-latest.sql'

# sync retrieved hashes to the IPFS Master node for distribution to peers
scp -rp /mnt/bigdisk/database2/ipfs [redacted] /home/ipfs/ipfs-from-chain

# added crontab on remote instead
ssh [redacted] './import_ipfs.sh'
```

Execute the scraper db maker

dump database to file

Transfer database to dedicated database server

Copy found IPFS data to the main swarm server

# Retrieving Asset Data from Ravencoin (What the Scraper DB Forker looks like)

Step 1. Get all Ravencoin Asset Names via RPC call listassets

Step 2. for each asset run RPC call getassetdata {asset} write to file '\$nth' term.

Step 3. For \$nth term files download Ipfs objects if one indicated in JSON output from Ravencoin chain.

Step 4. Insert records to the database using the variables collected in steps 1 thru 3.

```
root@bridge:/mnt/bigdisk/database2
#!/bin/bash
# Author: push
# Date: 08/01/2019
# Descriptin: Create database from raven-cli asset data for
# explorer asynchronous db access

raven-cli listassets > allassets.txt RETRIEVE ALL ASSET NAMES FROM RAVENCOIN RPC AND WRITE TO FILE

cat allassets.txt | sed 's/"/\\g' | sed 's/,/\\g' | sed "s/\\|/\\g" | sed 's/\\[/\\g' > allassets_filtered.txt FILTER ASSETS IN FILE

cat allassets_filtered.txt PRINT ASSETS TO TERMINAL

i=0 LOOP THRU ALL ASSETS
for asset in `cat allassets_filtered.txt`; do
    echo "the next asset is $asset"
    raven-cli getassetdata $asset > "processed/$i.json" RAVEN RPC CALL GET ASSET DETAILS THEN WRITE TO FILE

    echo "asset data is:\n"
    cat "processed/$i.json"
    echo "we make call to commit to the database" DOWNLOAD IPFS OBJECT IF EXISTS CREATE DATABASE SQL IMPORT DATA
    php downloadipfsobjects.php "processed/$i.json"

    i=$((i + 1)) GOTO NEXT OBJECT
done
```

# What does the deceptively named downloadipfsobjects.php file do??

```
if ($hasipfs == "1") {
    echo "Constructing URL Download params\n";

    #deprecated replaced by geturl
    $asseturl="https://gateway.lavenland.org/ipfs/" . $assetfiledecode->ipfs_hash;

    print "downloading file\n";

    $old_path = getcwd();
    chdir('/mnt/bigdisk/database2/');
    $outpath = shell_exec("./ipfsdownload sh $assetfiledecode->ipfs_hash");
    chdir($old_path);

    // Lets check the mime type notice this is inside this segment because we have many files existing
    // if you want to get the mime type of new additions then place it outside this function
    $file_to_check = "ipfs/" . $assetfiledecode->ipfs_hash;
    echo "CHECKING FILE MIME IS $file_to_check";
    $mime = mime_content_type("$file_to_check");
    echo "detected mime type is $mime";

    $insertipfs_hash="TRUE";

    // Create IPFS objects database for IPFS named assets attached

    $sql3 = "INSERT INTO asset_ipfs (name, ipfs_hash, json_file, mime_type)
VALUES ('$assetfiledecode->name', '$assetfiledecode->ipfs_hash', '$fileobject', '$mime')";

    // Create Asset Filesystem Objects Name lookup database entries

    $sql4 = "INSERT INTO asset_json (name, filename)
VALUES ('$assetfiledecode->name', '$fileobject')";

    // TAKE CARE OF ASSET NAMES from Havencoin RPC getassetdata JSON OUT
    $sql = "INSERT INTO assets (name, amount, units, reissuable, has_ipfs, ipfs_hash, ipfs_mime)
VALUES ('$assetfiledecode->name', '$assetfiledecode->amount', '$assetfiledecode->units', '$assetfiledecode->reissuable', '$assetfiledecode->has_ipfs', '$assetfiledecode->ipfs_hash', '$mime')";

    // Retrieve all Coin Holdings of the Asset via RPC call

    $assetname = ($assetfiledecode->name);
    $assetbyaddresses = $ravencoin->listaddressesbyasset("$assetname");
    //print_r($assetbyaddresses);

    // coin, amount
    echo "Inserting Asset Address Records";
    print "\n";
    foreach($assetbyaddresses as $key => $value) {
        echo "Address: $key as balance $value";
        echo "\n";
    }
    $sql2 = "INSERT INTO assetaddresses (assetname, address, amount)";
}
```

Quite a bit! Basically though it just takes data from json file output from raven, downloads any attached IPFS object, catalogues it in a variable and then commits all the data into the database.

# A Ravencoin Asset Explorer Database Schema

- 🌀 **Table 'asset\_ipfs'** (stores ravencoin ipfs objects)
- 🌀 **Table 'asset\_json'** ( stores ravencoin json get assetdata rpc call)
- 🌀 **Table 'assetaddresses'** (stores all ravencoin token holder amounts + asset names from rpc listaddressesbyasset rpc call)
- 🌀 **Table 'assets'** (stores all ravencoin asset details by name relating to other databases)

```
MariaDB [assetexplorer]> describe asset_ipfs;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NONE	auto_increment
name	varchar(32)	NO		NONE	
ipfs_hash	varchar(50)	NO		NONE	
json_file	varchar(50)	NO		NONE	
mime_type	varchar(127)	NO		NONE	

5 rows in set (0.00 sec)

```
MariaDB [assetexplorer]> describe asset_json;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NONE	auto_increment
name	varchar(32)	NO		NONE	
filename	varchar(50)	NO		NONE	

3 rows in set (0.00 sec)

```
MariaDB [assetexplorer]> describe assetaddresses;
```

Field	Type	Null	Key	Default	Extra
id	int(12)	NO	PRI	NONE	auto_increment
assetname	varchar(32)	NO		NONE	
address	varchar(35)	NO		NONE	
amount	bigint(16)	NO		NONE	

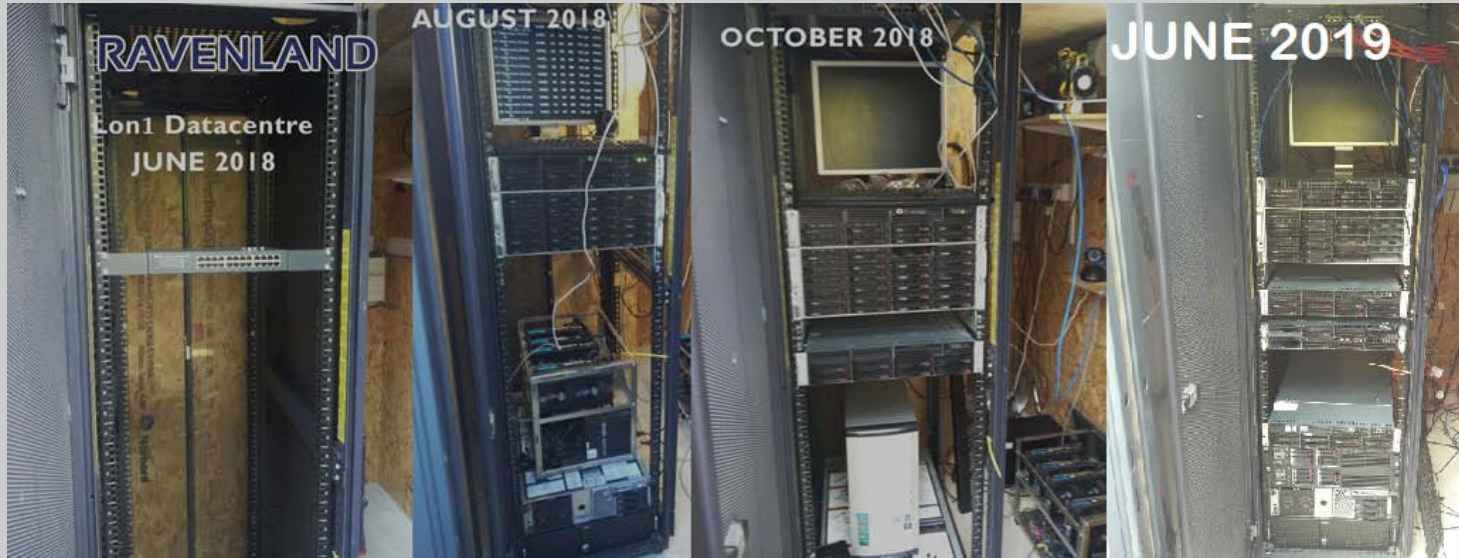
4 rows in set (0.01 sec)

```
MariaDB [assetexplorer]> describe assets;
```

Field	Type	Null	Key	Default	Extra
id	int(12)	NO	PRI	NONE	auto_increment
name	varchar(32)	NO		NONE	
amount	bigint(16)	NO		NONE	
units	int(12)	NO		NONE	
has ipfs	int(12)	NO		NONE	
ipfs_hash	varchar(50)	NO		NONE	
reissuable	int(11)	NO		NONE	
ipfs_mime	varchar(127)	NO		NONE	

8 rows in set (0.00 sec)

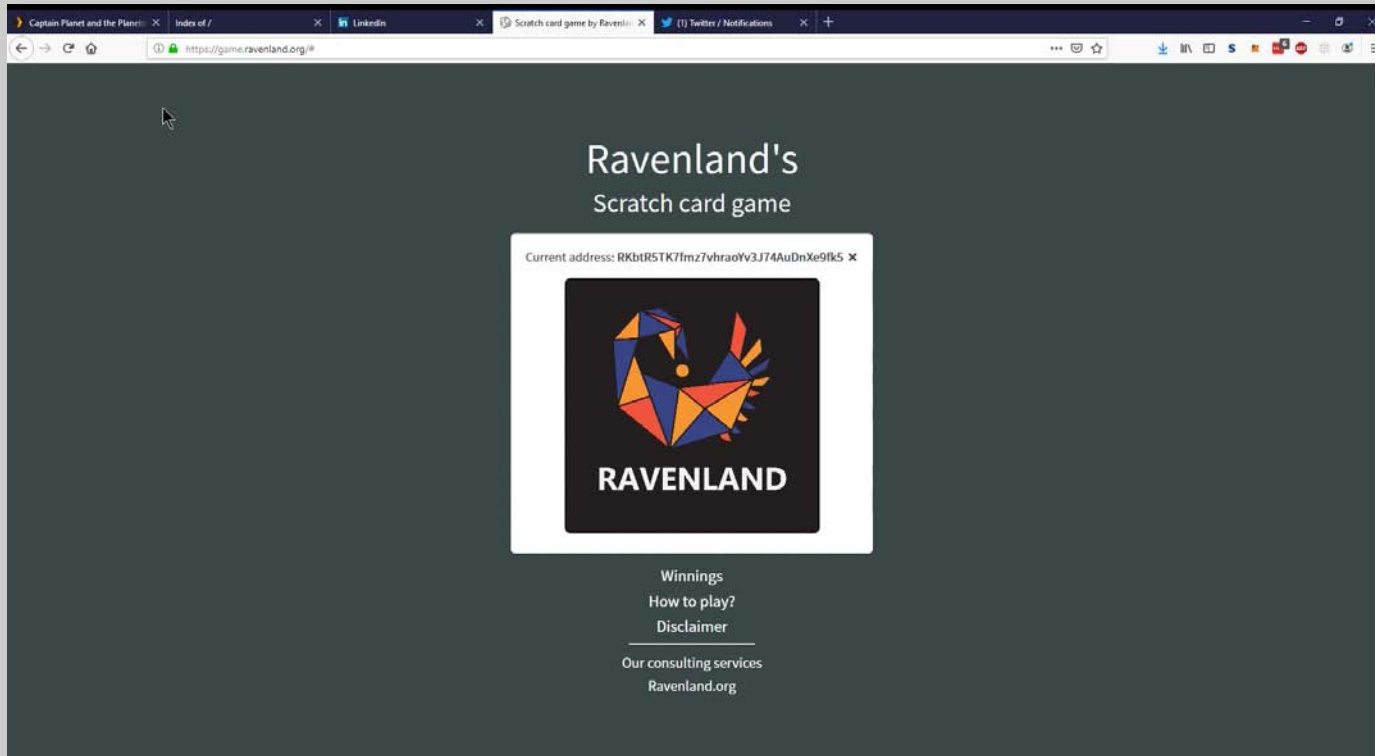
# Ravenland LON1 Facility Changes over the last year or so



A growing set of Rich IPFS + Ravencoin Services



# Taking it further: SCALA Ravenland API powered Scratch card Game



```

adam@terminalserver -
id to insert update transaction is 9509 - 09bda8515eacccce04400cfd3d68a6d260cb4aa31b615d3931cac87717035
=====
paying assets to RQKZkE7j1q2f8BZTHB5gLn1jYRP4ERAYbk 0.007 RVN
Making call to RPC
id to insert update transaction is 9516 - d6b8ed30a3bcd27f525279c132d8e45de70f155987dbcfbbdd784ea7c21a2fa3
=====
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!Outputting total ASSET winners at this moment..
=====
paying assets to RQKZkE7j1q2f8BZTHB5gLn1jYRP4ERAYbk 10 GOTO/RAVENCOIN.NETWORK
Making call to RPC
id to insert update transaction is 9506 -
45e37elcc3e323d3d669e9206565af735dfd313d5daa409545ddd0fb57bb65bd
=====
paying assets to RMGKr7j1q2f8BZTHB5gLn1jYRP4ERAYbk 500000 CATE
Making call to RPC
id to insert update transaction is 9507 -
a5a2c38a280dc4b7d43066e162957f4c060d09c262ae5bc41f541rldf8de509
=====
paying assets to RWfFjXHWjyyFLyFd2kr7FUF5ocETfXeSgQ 500000 CATE
Making call to RPC
id to insert update transaction is 9508 -
a149e0503170a54492aa93fbc319f72ea4cad6e0dbafa0b20b34e8feed7305
=====
paying assets to RQKZkE7j1q2f8BZTHB5gLn1jYRP4ERAYbk 10 HANNUKAR_GELT
Making call to RPC
id to insert update transaction is 9510 -
57a650d3D1cle29ca30dc190ed15ae1aa9d935346blb147a59037d69247lbbc
=====
paying assets to RWfFjXHWjyyFLyFd2kr7FUF5ocETfXeSgQ 10 SLAP
Making call to RPC
id to insert update transaction is 9511 -
6df96f2e03b660cd7aea7d687c1650e4182f31d15e9453f5dbe4508d26c4a4b4
=====
paying assets to RQKZkE7j1q2f8BZTHB5gLn1jYRP4ERAYbk 10 STROBFX.COM
Making call to RPC
id to insert update transaction is 9512 -
e43708960c13c75fe8acdcd950a50be679a3d20b75d51c99cae9bde75e074b1
=====
paying assets to RLUpRwVpHMj8VgbjVGGFtpRCn87ZMNB1C 10 0.YESRVN.COM
Making call to RPC
id to insert update transaction is 9513 -
leld80fd2e013a9165b1b28ba2a721880e35181cab4948d3053459cbb532fde3

```

```
if is_distributed=0 then payout
customer, carry returned
transactionid to database && set
is_distributed=1.
```

# Ultimate Goals

- Create products. Show **demonstrability**.
- Create products **before** raising money.
- Have an **honest funding model** to forward the companies objectives as a **common stock for all** to bring forward a technology for our new future.
- **Give back to the investor.** Create a real 'RavenLand' token that directly represents the companies economies of worth.
- **Give the community an open-source software vehicle** for commercial and industrial retail with the **power to end captive markets and societies as we know them** and restore a fundamental economic balance in world-trade.
- **Restore control to the public** of **their** communities and **their** services.

# Important References

- <https://www.ravenland.org> ( Official Ravenland website)
- <https://www.ravencoin.com> ( Official Ravencoin website)
- [RavencoinOS website https://www.ravenland.org/os](https://www.ravenland.org/os) / [www.ravencoinos.org](https://www.ravencoinos.org)
- [Ravencoin https://github.com/RavenProject/Ravencoin/releases](https://github.com/RavenProject/Ravencoin/releases)
- [Ravencoin Bootstrap Tools https://github.com/ravenlandpush/ravencoin-ipfs-bootstrap-tools](https://github.com/ravenlandpush/ravencoin-ipfs-bootstrap-tools)
- [Ravencoin Asset Explorer https://assetexplorer.ravenland.org/](https://assetexplorer.ravenland.org/)
- [Ravencoin Bootstrap Services https://bootstrap.ravenland.org/](https://bootstrap.ravenland.org/)
- [Automatic IPFS Installer for Linux https://github.com/ravenlandpush/RavencoinIPFSInstaller](https://github.com/ravenlandpush/RavencoinIPFSInstaller)
- [How to Airdrop Assets https://github.com/ravenlandpush/RavencoinAssetDistributor](https://github.com/ravenlandpush/RavencoinAssetDistributor)
- [Ravencoin Multi-Faucet https://faucet.ravenland.org/](https://faucet.ravenland.org/)
- [How to Install IPFS https://docs.ipfs.io/introduction/install/](https://docs.ipfs.io/introduction/install/)
- [Ravenland IPFS Gateway for Ravencoin https://gateway.ravenland.org/ipfs/](https://gateway.ravenland.org/ipfs/)
- [Ravenland IPFS Upload Service https://interplanetary.ravenland.org/](https://interplanetary.ravenland.org/)
- [Ravenland Game https://game.ravenland.org](https://game.ravenland.org)
- [Ravenland Mirror - https://mirror.ravenland.org](https://mirror.ravenland.org)