OCI | WE ARE SOFTWARE ENGINEERS.

# A Developer's View of EOSIO

Presented by Phil Mesnier

# OVERVIEW

- Contextualizing "developer"

- Overview of EOS and EOSIO

- Tools provided with EOSIO

- Smart contracts and decentralized apps

objectcomputing.com

# WHAT DOES "A DEVELOPER'S VIEW" MEAN?

- Looking at the EOSIO services from a developer's perspective

- Core developer?

- Web/front-end developer?

- DApp/contract developer?

- Decentralized app projects can engage all three

# CORE DEVELOPER

- Works on C++ code most commonly found in the libraries and plugins directories of eosio/eos

- Most developers will not have to work at this level

- May be necessary if adding new capabilities to the blockchain

# WEB/FRONT-END DEVELOPER

- Works on application code "above" the API

- Uses tools such as those found in eosio/eosjs

- May also create non UI system services that also live above the API

- Unfortunately, not an area I can say much about

# CONTRACT DEVELOPER

- Works on code "below" the API

- Uses C++ and adapters such as those found in eos/contracts/eoscpp

- Defines actions that interact with the blockchain

# Overview of EOSIO

objectcomputing.com

# EOS and EOSIO

- EOS is a token.

- EOSIO is the software platform for deploying decentralized applications.

# Delegated Proof of Stake (DPoS)

- Block creation responsibility shared amongst 21 well-known, voted-in block producers

- Standby block producer pool has over 100 candidates.

- Voting is continuous.

# PERFORMANCE

# DECENTRALIZATION

This shows that DPoS in fact increases the overall block producer decentralization, whereas with bitcoin and Ethereum, mining pools have become effective central authorities with the top 5 producer pools controlling over 70% of the block production.
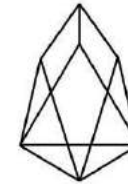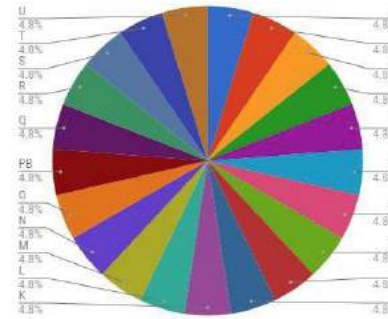


EOS

Source: EOS.IO WhitePaper
21 Block Producers
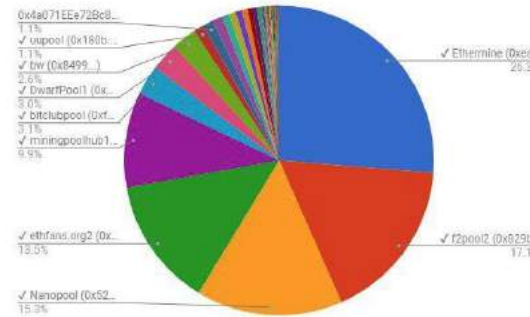
ETHEREUM

Source: etherchain.org
Top Miners over last 24 hrs
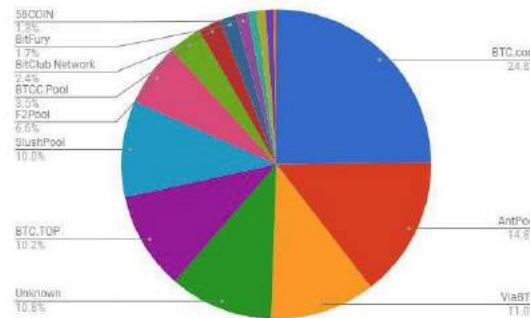
BITCOIN

Source: Blockchain.info
Hashrate Distribution - 4 Days

Brought to you by EOSoCal.io - A Block Producer Candidate          03/05/2018

objectcomputing.com

# THE SOFTWARE BUNDLE

- Download available from GitHub. Package is eosio/eos

- git clone https://github.com/EOSIO/eos — recursive

- A lot of external dependencies, particularly on LLVM and Boost

- Supports Linux and MacOS

- Does not support Windows out of the box

# THE BRANCHES IN git

- Branch "master" contains the latest patch release

- Branch "release" contains the latest major release

- Branch "develop" contains the upcoming patch release

**objectcomputing.com**

# IMPORTANT TOP-LEVEL DIRECTORIES

Docs and contacts top list

# BUILDING THE BUNDLE

- good: use ./eosio_build.sh script

- It will guide you through resolving your dependencies.

- better: use an IDE, for example XCode or CLion

**objectcomputing.com**

# A SUCCESSFUL BUILD!



```
EOSIO has been successfully built. 00:13:06

To verify your installation run the following commands:

/usr/local/bin/mongod -f /usr/local/etc/mongod.conf &
cd /tao_builds/phil/projects/eos/atl/eos/build; make test

For more information:
EOSIO website: https://eos.io
EOSIO Telegram channel @ https://t.me/EOSProject
EOSIO resources: https://eos.io/resources/
EOSIO Stack Exchange: https://eosio.stackexchange.com
EOSIO wiki: https://github.com/EOSIO/eos/wiki
```

objectcomputing.com

# EOSIO Tools

objectcomputing.com
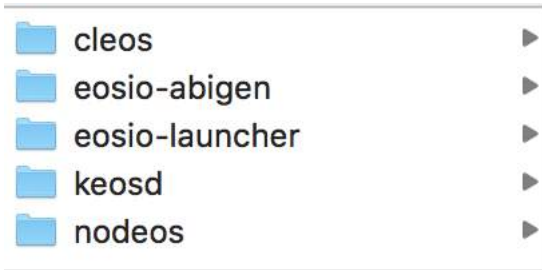
# EOSIO PROGRAMS



- Found in eos/build/programs

- Could be installed to usr/local/bin

# nodeos

- The heart of any eosio-derived network

- Single threaded (mostly)

- Modular construction using plugins

# AVAILABLE PLUGINS

- In directory eos/plugins

- Several plugins separate function from API

# RUNNING NODES

- A single nodeos may be run as a "network" to perform most testing necessary during application development

- Start nodeos from the command line

- Runs forever, builds a block archive

- Can be configured to join a running network, must synchronize its block database with the rest of the network.

```
Application Command Line Options:
  -h [ --help ]                        Print this help message and exit.
  -v [ --version ]                     Print version information.
  --print-default-config              Print default configuration template
  -d [ --data-dir ] arg               Directory containing program runtime
                                       data

  --config-dir arg                     Directory containing configuration
                                       files such as config.ini

  -c [ --config ] arg (=config.ini)    Configuration file name relative to
                                       config-dir

  -l [ --logconf ] arg (=logging.json) Logging configuration file name/path
                                       for library users
```

**objectcomputing.com**

# eosio-launcher

- nodeos configuration for multiple nodes requires many similar values and some different ones

    Each node has a listen socket or two that is unique

    Each node also connects to some peers based on desired network topology

- The launcher simplifies creation of the config files using simple network definition lists

- Deploys files and launches nodes using ssh and scp commands

# cleos

- Command line interface to the blockchain

- Performs actions, from checking the status of a single node to creating accounts, loading contract, and invoking actions

- Will automatically load an instance of keosd if needed

objectcomputing.com

# cleos AND COMMAND STRUCTURE

- Parameter list varies based on what command is being issued

- Most common use case is invoking contract actions

- Most common arguments are nodeos host/port

- Example: cleos create account eosio eosio.token EOS7ijWCB

# keosd

- A keystore manager

- Should only listen to localhost connections

- Can be run separately or started on demand by cleos

# eosio-abigen

- Generates ABI files for contracts based on C++ definitions

# EOSIO Contracts

objectcomputing.com

# CONTRACT ESSENTIALS

- Written in C++

- Compiled by WASM

- Associated with accounts on the blockchain

- Run duration is limited

- Every invocation reloads/validates/initializes the contract

# ACTIONS AND TABLES

- Actions are functions that may be triggered by an invocation

- Tables are used to persist data on the blockchain

- Consider the Tic-Tac-Toe contract

```json
    ],
    "actions": [{
        "name": "create",
        "type": "create",
        "ricardian_contract": ""
    },{
        "name": "restart",
        "type": "restart",
        "ricardian_contract": ""
    },{
        "name": "close",
        "type": "close",
        "ricardian_contract": ""
    },{
        "name": "move",
        "type": "move",
        "ricardian_contract": ""
    }
    ],
    "tables": [{
        "name": "games",
        "index_type": "i64",
        "key_names": [
            "challenger"
        ],
        "key_types": [
            "name"
        ],
        "type": "game"
    }
    ],
```

# CREATE OR GENERATE AN ABI

- "Application Binary Interface" is a JSON definition of all the types and values that are passed in invocations or stored in tables

- Types are primitives or structs. Type definitions may include base types defined external to the ABI

- EOSIO contains a common library of type definitions

- ABI files can be generated from the C++ header files

# TIC-TAC-TOE FULL ABI

```json
{
    "___comment": "This file was g
    "version": "eosio::abi/1.0",
    "types": [],
    "structs": [{
        "name": "game",
        "base": "",
        "fields": [{
            "name": "challenger",
            "type": "name"
        },{
            "name": "host",
            "type": "name"
        },{
            "name": "turn",
            "type": "name"
        },{
            "name": "winner",
            "type": "name"
        },{
            "name": "board",
            "type": "uint8[]"
        }
        ]
    },{
        "name": "create",
        "base": "",
        "fields": [{
            "name": "challenger",
            "type": "name"
        },{
            "name": "host",
            "type": "name"
        }
    ]
```

```json
    },{
        "name": "restart",
        "base": "",
        "fields": [{
            "name": "challenger",
            "type": "name"
        },{
            "name": "host",
            "type": "name"
        },{
            "name": "by",
            "type": "name"
        }
    ]
    },{
        "name": "close",
        "base": "",
        "fields": [{
            "name": "challenger",
            "type": "name"
        },{
            "name": "host",
            "type": "name"
        }
    ]
```

```json
    },{
        "name": "move",
        "base": "",
        "fields": [{
            "name": "challenger",
            "type": "name"
        },{
            "name": "host",
            "type": "name"
        },{
            "name": "by",
            "type": "name"
        },{
            "name": "row",
            "type": "uint16"
        },{
            "name": "column",
            "type": "uint16"
        }
    ]
    }
    ],
```

```json
    "actions": [{
        "name": "create",
        "type": "create",
        "ricardian_contract": ""
    },{
        "name": "restart",
        "type": "restart",
        "ricardian_contract": ""
    },{
        "name": "close",
        "type": "close",
        "ricardian_contract": ""
    },{
        "name": "move",
        "type": "move",
        "ricardian_contract": ""
    }
    ],
    "tables": [{
        "name": "games",
        "index_type": "i64",
        "key_names": [
            "challenger"
        ],
        "key_types": [
            "name"
        ],
        "type": "game"
    }
    ],
    "ricardian_clauses": [],
    "error_messages": [],
    "abi_extensions": []
}
```

# RICHARDIAN CONTRACTS

- A response to the notion that "code is law"

- Provided for every action, written in English

- Structured to be both human and machine readable

- Enable arbitration by codifying "intent"

- Enforce the notion that "intent is law"

# ACCOUNT NAMES

- Why the weird restrictions? only 'a' - 'z', '1' - '5', '.'

- That is 32, or 2^5, characters

- So a name is composed of 12 and 1/2 characters making a 64 bit value.

**objectcomputing.com**

**OCI** | WE ARE SOFTWARE ENGINEERS.

## CONNECT WITH US

📞 1+ (314) 579-0066

🐦 @objectcomputing

🔍 objectcomputing.com

**objectcomputing.com**