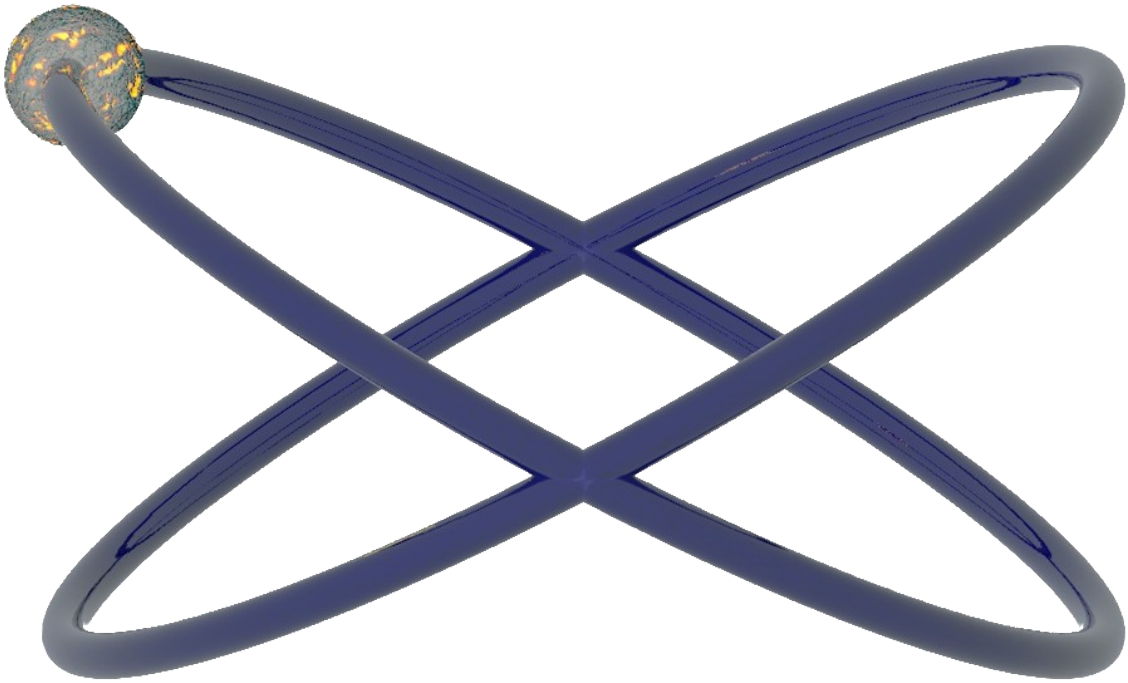
Runebase(RUNES)



This is not a Whitepaper
0.0.1

Table of Contents

1. **Legal notice**

2. **Introduction**

3. **Reward System**
   2.1 Type: Proof of Stake
   2.2 BlockTime: ~2 min
   2.3 BlockReward: 100 RUNES
   2.4 Transaction Fees
   2.5 Max Coins

4. **Consensus**
   4.1 The Problem: Proof-of-Work
   4.2 The Solution: Proof-of-Stake

5. **Economic model**

6. **Contracts & EVM integration**
   6.1 EVM Integration
   6.2 Account Abstaction Layer
   6.3 Gas model

7. **Future directions**
   7.1 Adoption
   7.2 Development
   7.3 Partners

8. **Opportunities**

Abstract. Runebase is a smart contract platform that uses proof-of-stake to validate transactions, proof-of-stake has significant performance advantages compared to proof-of-work solutions.

Runebase is a combination of Bitcoin Core, Proof of stake and the Ethereum Virtual Machine (EVM). Runebase Core, allows smart contracts to execute on a Proof-of-Stake consensus model. The ecosystem provides familiar enviroment for smart contracts and Decentralized Application Developers.

The Runebase Foundation's goal is to make sure expenses and funding for future development are covered. Another RUNES goal is to maintain compatibility with existing processes from Bitcoin and Ethereum, to be secure as possible, to be easily used by individuals, businesses and developers.

Runebase is secured by Proof-of-Stake 3.0 (POS 3) consensus protocol and is integrated with a smart contract platform. Smart contracts are executed as a part of a Unspent Transaction Output (UTXO), which is part of the bitcoin transaction model. Thus, the following advantages are maintained:

1) Compatibility with existing bitcoin workflows
2) Privacy aspect of the Bitcoin UTXO model
3) Bitcoin's UTXO model is scalable for the long term
4) Integration with existing models
5) A implementation which has been proven to be secure for over 11 years

1. Legal notice

This paper is intended for informational purposes only. It is not intended to be investment advice, solicitation of any kind nor an endorsement. Any decisions or actions taken on the basis of information presented in this whitepaper, Runebase websites or other content is done at your own risk and discretion.

2. Introduction

Runebase is a community-driven cryptocurrency, we define being community-driven as putting people first. By educating, increasing people's awareness about cryptocurrency and reaching out to new community members we can rally up support for future goals.

3. Reward system

2.1 Type: Proof of Stake

Runebase is secured by Proof-of-Stake 3.0 (POS3). To be eligible for staking, the funds of your account require 500 confirmations since the last transaction of the funds. Active stakes will be locked out for 500 blocks, thus you won't be able to spend the funds that are staked during this period. The locked funds are returned after the 500 block lockout period.

2.2 BlockTime: ~2 min

The network aims to produce 1 block approximatly every 2 minutes

2.3 BlockReward: 100 RUNES

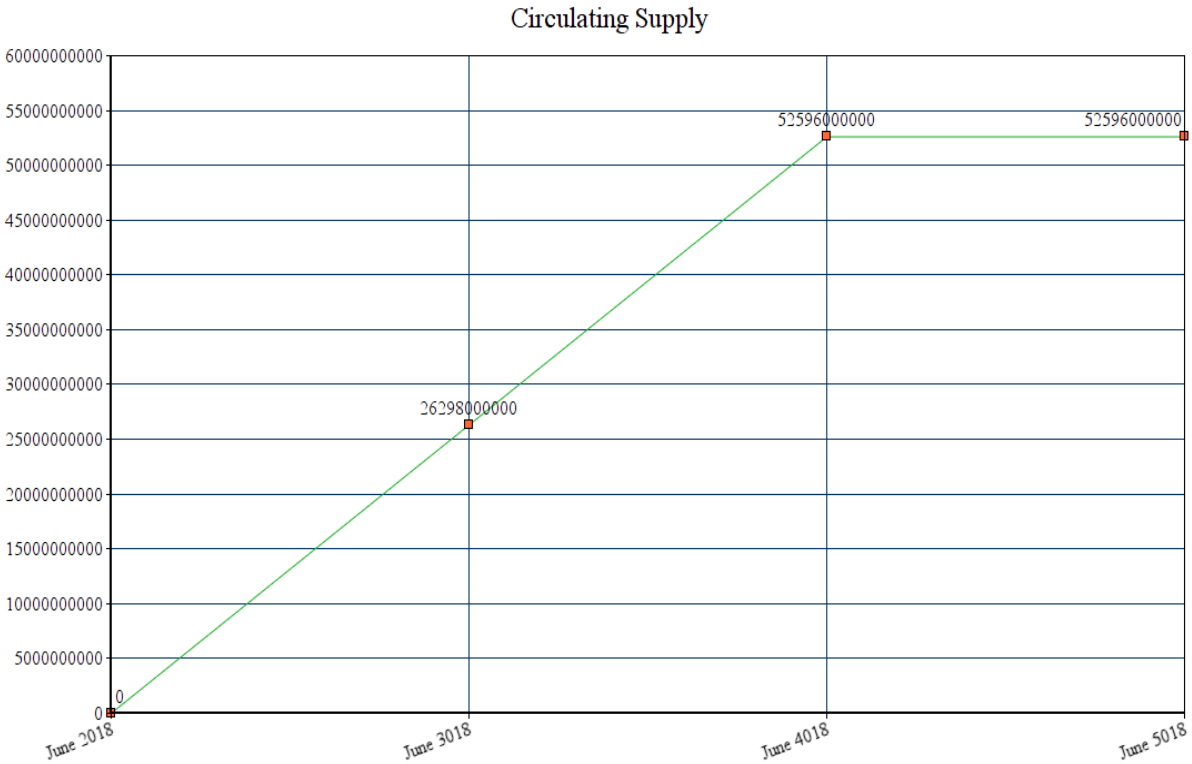
Every block contains a reward of 100 RUNES. There is no halving on the Runebase Network. Block producers will receive their reward in chuncks of 10 RUNES which are rewarded after 500 blocks

2.4 Transaction Fees

The minimum required transaction fee is 0.004 RUNES/kb
Transaction Fees are rewarded to the block producer.

2.5 Max Coins

The max amount of coins that will ever be produced is 52596000000, however to reach the maximinum coins is a process that will take over 2000 years. Every block produces 100 coins until max supply is reached, so there is plenty of room for people to enjoy the benefits from the reward pool.



4. Consensus

4.1 The Problem: Proof-of-Work

Cryptocurrency mining operations require high-powered computers to perform proof of work calculations. This requires many specialized computer parts, including graphics processing units (GPUs), application specific integrated circuits (ASICs), and field programmable gate arrays (FPGAs). These are the very same components that are required by the world's artificial intelligence researchers, computer scientists, medical researchers, gamers and game designers, animators, graphic designers, and video editors, amongst others. Due to the increasing popularity and profitability of cryptocurrency mining, these critical computing components are becoming scarce. Miners are currently using this hardware to ensure network consensus via proof-of-work, however it could be used for the betterment of society in many other ways. Those who need this computing hardware are forced to settle for suboptimal hardware, pay inflated prices on secondary markets, or pay exorbitant fees to rent the necessary hardware on cloud computing platforms. A large portion of the world's critical computing hardware is being used by miners to everyone else's detriment. To grasp how much computing power is tied up in the Bitcoin network, it is helpful to compare the Bitcoin network to the world's most powerful super computer, Sunway TaihuLight.(14) One metric to measure computing power is PETAFLOPS, or quadrillion floating point operations per second. Sunway TaihuLight can perform bursts of up to 93 PETAFLOPS.(15) The Bitcoin network currently outputs 80,704,290.84 PETAFLOPS to solve the proof-of-work problem and secure the network.(16) This means the Bitcoin network sucks up computing power equivalent to nearly 900,000 copies of the world's most powerful super computer. One can only imagine what could be achieved if that scale of computing power was redirected perhaps towards areas of computationally intensive research, such as deep learning. The computing power of Bitcoin only continues to grow, and this problem will only get worse, as evidenced by the network's exponentially, and nearly monotonically, increasing hashrate.**(2)(3)**

4.2 The Solution: Proof-of-Stake

The advantages of the Proof of Stake algorithm are energy efficiency and security. A large number of users are encouraged to run nodes because it's easy and affordable. This along with the randomization process also makes the network more decentralized, since mining pools are no longer needed to mine the blocks.

5. Economic model

The Foundation functions as an umbrella under which all expenses and funding for the development of Runebase are managed. This means full transparency towards the community on the costs and funding of the project, but also the community's participation on how we want or can fund future developments. The Foundation expects to have a revenue stream to maintain its sustainability and to ensure it's goals are achieved. any surplus cash flow may be used to support technology development or as donations to charitable causes.

6. Contracts & EVM integration

The Ethereum Virtual Machine is a stack-based virtual machine with a 256-bit machine word. Smart contracts which run on Ethereum use this virtual machine for their execution. The EVM is designed for Ethereum's blockchain, and assumes that all value transfer will be done using an account-based method. Runebase is based on Bitcoin's blockchain design, however, and uses the UTXO-based model. Runebase has an Account Abstraction Layer which translates the UTXO-based model to an account-based interface for the EVM to use. As well as this there is an additional Blockchain Interface so that the EVM can directly access various details about the Runebase blockchain. **(1)**

6.1 EVM Integration

All transactions in Runebase use the Bitcoin Scripting Language, just like Bitcoin. In Runebase however, there are 3 new opcodes.

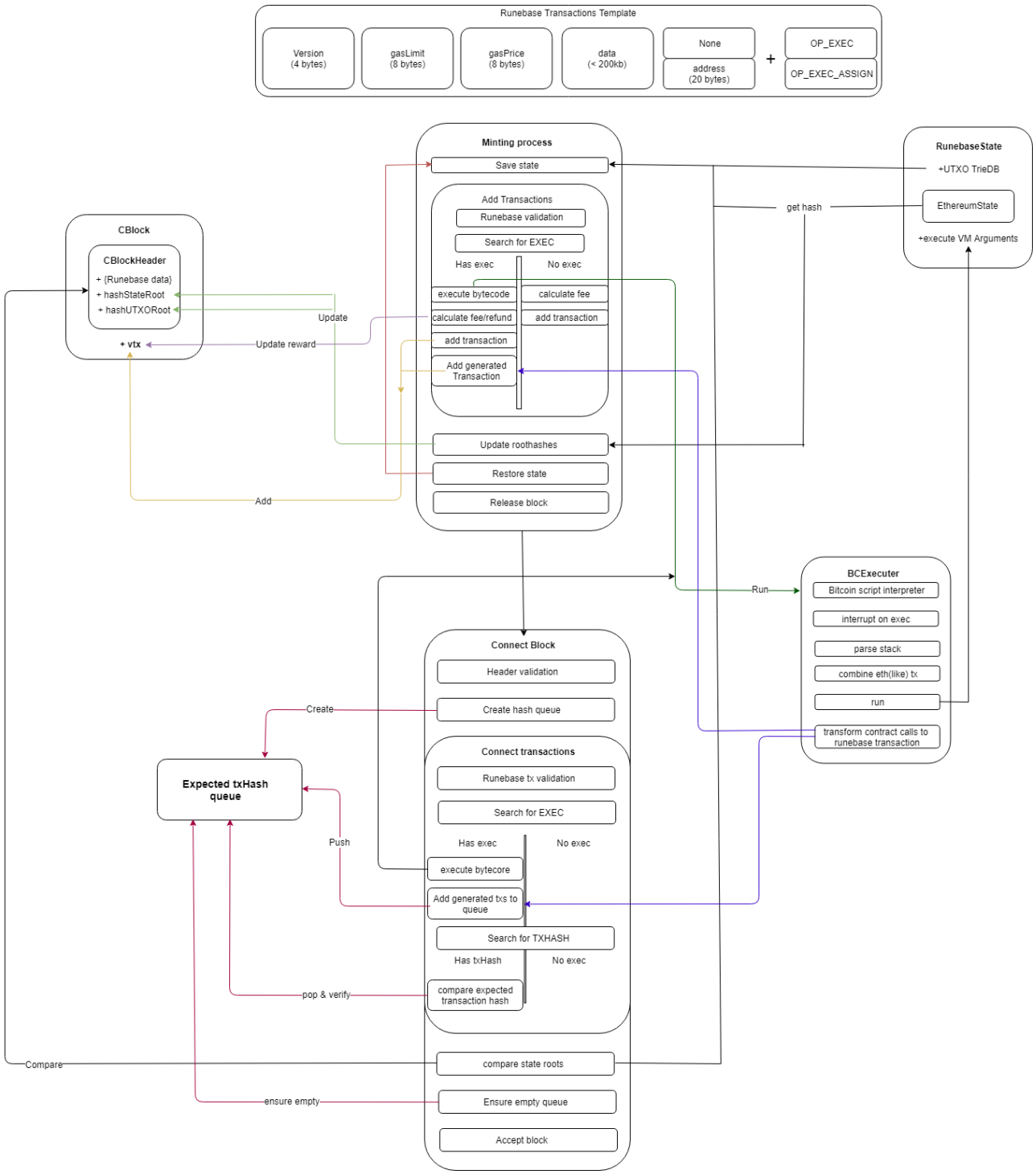
1. OP_EXEC - This opcode will trigger special processing of a transaction (explained below) and will execute the EVM bytecode passed to it.

2. OP_EXEC_ASSIGN - This opcode will also trigger special processing like OP_EXEC. This opcode is passed a contract address and data to give the contract. It will then execute the contract's bytecode while passing in the given data (given as CALLERDATA in EVM). This opcode is also used for giving money to a smart contract.

3. OP_TXHASH - This opcode is used to reconcile an odd part of the accounting abstraction layer. It simply pushes the transaction ID hash of the current transaction being executed. **(1)**

6.2 Account Abstaction Layer

The Ethereum Virtual Machine is designed to function on an account-based blockchain. Runebase however, being based on bitcoin, uses a UTXO-based blockchain. To handle this, Runebase contains an Account Abstraction Layer which will allow the Ethereum Virtual Machine to function on the Runebase blockchain without significant modifications to the virtual machine nor existing Ethereum contracts. The EVM account model exposed to smart contract programmers is fairly simple. There are operations that can check the balance of the current contract and other contracts on the blockchain, and there are operations which can send money (attached to data) to other contracts. Although these actions seem fairly basic and minimalistic, they are not trivial to do within the UTXO-based Runebase blockchain. Thus, the Account Abstract Layer's implementation of these operations may be more complex than expected. First off, when a smart contract is deployed to the Runebase blockchain it is assigned and callable by its transaction hash. A newly deployed contract's balance will also be zero. There is currently no protocol in Runebase that allows a contract to be deployed with a non-zero balance. In order to send funds to a contract, a transaction will be created which uses the OP_EXEC_ASSIGN opcode. The output script which sends money to the contract looks similar to this: 1; gas limit for the transaction 100; gas price in Runebase satoshis 0xF012; data to send the contract (using the Solidity ABI) 0x1452b22265803b201ac1f8bb25840cb70afe3303; ripemd-160 hash of contract txid OP_EXEC_ASSIGN This transaction script is fairly simple and hands off most of the transaction processing to the OP_EXEC_ASSIGN opcode. The actual value amount given to the contract from this (assuming 17 there is not an out-of-gas condition) is OutputValue - GasLimit. The exact details of the gas mechanism will be discussed later. When this output is added to the blockchain, it becomes an output that belongs to the contract's account. This output's value will be reflected in the balance of the contract.The balance is simply the sum of the outputs which are spendable by the contract. **(1)**


Runebase Transactions Template
Version (4 bytes)
gasLimit (8 bytes)
gasPrice (8 bytes)
data (< 200kb)
None
address (20 bytes)
+
OP_EXEC
OP_EXEC_ASSIGN
Minting process
Save state
Add Transactions
Runebase validation
Search for EXEC
Has exec
No exec
execute bytecode
calculate fee
calculate fee/refund
add transaction
add transaction
Add generated Transaction
Update roothashes
Restore state
Release block
RunebaseState
+UTXO TrieDB
EthereumState
+execute VM Arguments
get hash
CBlock
CBlockHeader
+ {Runebase data}
+ hashStateRoot
+ hashUTXORoot
+ vtx
Update
Update reward
Add
BCExecuter
Bitcoin script interpreter
interrupt on exec
parse stack
combine eth(like) tx
run
transform contract calls to runebase transaction
Run
Connect Block
Header validation
Create hash queue
Create
Expected txHash queue
Connect transactions
Runebase tx validation
Search for EXEC
Has exec
No exec
execute bytecore
Add generated txs to queue
Search for TXHASH
Has txHash
No exec
compare expected transaction hash
Push
pop & verify
Compare
compare state roots
Ensure empty queue
ensure empty
Accept block

(1)

5.3 Gas model

The gas concept can be summarized by saying that each EVM opcode executed has a price, and each transaction has an amount of gas which can be spent. Whatever amount of gas remains after the transaction is complete will be refunded back to the sender. Also if the amount of gas required to execute a contract exceeds the amount of gas available to a transaction, then the transaction's actions and state changes are reverted. This means any permanent storage that has been modified will be reverted to its original state, and any spending of contract funds will be reverted so that they are not spent. Even though all of this state is reverted, all of the gas of the transaction is consumed and given to the miner processing it. This is because the computing resources have already been spent by it's processing, so even though it's not safe to cause any state changes on the blockchain, the processing power has been spent and should go to the miner for its effort. Although Runebase uses the gas model from Ethereum, it is expected that the gas schedule (gas price of each EVM opcode) will significantly differ from Ethereum. This is because in Runebase some operations are more expensive than in Ethereum, and some operations are cheaper. The exact values will be determined by looking at existing prices in Ethereum and comparing the amount of processing and blockchain resources required for each opcode in comparison to Runebase. When creating a contract funding or deployment transaction, the user specifies two specific items for gas. The first is the GasLimit, which is how much gas can be consumed by this contract execution. The second is the GasPrice, which is the exact price of each unit of gas in Runebase satoshis. The maximum Runebase expenditure of a contract execution can thus be easily 22 computed by GasLimit multiplied by GasPrice. If this maximum expenditure exceeds the transaction fee provided by the transaction, then the transaction is considered invalid and will not be mined or processed. The remaining transaction fee after this maximum expenditure is subtracted is the Transaction Size Fee. This is analogous to the standard Bitcoin fee model. To determine the appropriate priority of a transaction, miners must now look at two variables. First, the transaction size fee should be appropriate for the total size of the transaction (usually determined by a minimum amount of coins per kilobyte formula). The second variable is, of course, the GasPrice of a contract execution. Together, proof-of-stake miners have a great degree of choice in choosing the most important and profitable transactions to process and include in a block. This allows the fee model to work like a free market, with miners and users optimizing for the best fee that suites their transaction's speed and the price they are willing to pay. **(1)**

7. Future directions

Below we set out goals. We have not set any dates for when we anticipate these to be completed at this time. We are constantly evolving and try to adjust to any given climate.

7.1 Adoption

The trustless nature of the blockchain is incredibly powerful, but we believe a bridge of social trust is still needed for us to achieve adoption. The adoption process plays a big part in our ability to grow and to expand our reach by listing on additional exchanges.

7.2 Development

Runebase will stay compatible with bitcoin, the latest bitcoin core software updates will be integrated into the the runebase core software. We will release a iphone wallet once the barriers are lifted which restrict us from doing so. The foundation will direct and fund the development that give partners the ability to build, grow, and create value for one another. As part of this process, All code produced by the Runebase foundation is made available as an open source project that can be leveraged to power new communities and add capabilities to existing ones. Our motto is underpromise and overdeliver.

7.3 Partners

Partnerships with complementary businesses can give access to new userbases. Partnerships often benefit both parties. Even though partnerships are often only for a specific goal, it can create long-lasting business relationships.

8. Opportunities

Runebase operates on a relatively young and growing market. Because the blockchain is still a new and developing technology, whose innumerable possibilities have not been tried so far, there is a chance that the market and its capitalization will grow. This may also result in an increase in RUNES price. One of the aspects affecting cryptocurrency development is directly linked to FIAT currencies, FIAT currencies weakening value, their lack of support in gold, growing inflation, as well as relatively small profits from deposits and bonds. These factors may cause investors to transfer their funds from traditional forms of investment to blockchain projects, including cryptocurrencies.

References

(1) https://whitepaperdatabase.com/qtum-whitepaper/

(2) S.A.R.L. Hash rate,2017 https://blockchain.info/charts/hash-rate?timespan=all

(3) https://www.savebitcoin.io/whitepaper