Bitcoin Wonder

A Bitcoin Creates Wonders

NOTE: the contents of this white paper are under active revision, comments are appreciated.

Index

1. Introducing Bitcoin Wonder

Bitcoin Wonder is a new decentralized ledger and a new chain generated by Bitcoin and Borderless on cross-chain protocol. Trading on Bitcoin Wonder is verified by witness node methodology. Bitcoin Wonder aims to provide help to those small-medium sized enterprises worldwide by freely offering them Bitcoin Wonder as support and create wonders for the globe, which is always the highest and most loyal pursuit of Bitcoin Wonder.

2. Bitcoin Wonder Market

A digital free market financial system that can facilitate trade in any asset class without introducing valueless middlemen or centralized issuers of assets. Bitcoin Wonder aims to provide help to those small-medium sized enterprises worldwide by freely offering them Bitcoin Wonder as support and create wonders for the globe, which is always the highest and most loyal pursuit of Bitcoin Wonder.

Built-in Decentralized Mining Pool

The techniques Bitcoin Wonder uses to enable a distributed mining pool with no central server could be integrated to make it quick and easy for most users to do some mining even as the difficulty increases. This would not be a requirement of the Bitcoin Wonder protocol, but would be supported by the network.

1) Security

a) 51% Denial of Service Resistance

All witness nodes have a financial incentive to validate chains. All miners have incentive to reject blocks that include a large number of 'never--before--seen' transactions and fees because it means someone is 'holding out' in an effort to collect fees or manipulate the network. Because most users can 'profitably' mine, all users will actively cooperate in preventing these kinds of manipulation attempts. As a result, the cost of a 51% DOS attack requires the attacker to subsidize the entire network and their competition which will increase the profitability of mining and thus make it more expensive to maintain the 51% Double Spend attack.

b) Encrypted Communications

All communication between nodes will be encrypted for two reasons: it will frustrate packet filtering, and it will make it harder to determine the origin of new transactions.

3. Technology Behind Bitcoin Wonder System

1) High Performance and Scalability

High performance blockchain technology is necessary for cryptocurrencies and smart contract platforms to provide a viable alternative to existing financial platforms.

To achieve this industry-leading performance, Bitcoin Wonder has borrowed lessons learned from the LMAX Exchange, which is able to process 6 million transactions per second. Among these lessons are the following key points:

1. Keep everything in memory.
2. Keep the core business logic in a single thread.
3. Keep cryptographic operations (hashes and signatures) out of the core business logic.
4. Divide validation into state-dependent and state-independent checks.
5. Use an object oriented data model.

By following these simple rules, future optimizations are expected to bring the performance of Borderless to levels similar to LMAX. It should be noted that the performance achieved by Borderless is highly dependent upon having a compatible transaction protocol. It would not be possible to achieve the same level of performance in a protocol where the Core Business Logic is run in a virtual machine that performs cryptographic operations and references all objects with hash identifiers. Blockchains are inherently single-threaded, and the performance of a single core of a CPU is the most limited and least scalable resource of all. Borderless is designed to get the most out of this single thread of execution

Background

A blockchain is a global ledger that orders transactions, whereby each transaction deterministically modifies a shared global state at a specified timestamp. The order in which transactions are processed can change the validity of other transactions. For example, you cannot withdraw money from your bank account until after your paycheck deposit has cleared. It becomes impossible to know whether or not a transaction is valid until after all prior transactions that impact a particular account have been processed. In theory, transactions for two unrelated accounts can be processed at the same time, provided that they do not share any common dependency. In practice, the cost of identifying which transactions are truly independent of each other on a ledger empowered by smart contracts with arbitrary conditions is intractable. The only way to be sure that two transactions are truly independent is by maintaining completely separate ledgers and then periodically transferring value between them. An analogy could be made to the performance trade offs in the design of Non-Uniform Memory Access (NUMA) vs Uniform Memory Access. In practice, Uniform Memory Access is much easier for developers to design for, and has lower costs. NUMA architectures are usually adopted as a last resort when building supercomputers or giant clusters. The computer industry has grown to realize that scaling performance through parallelism is nowhere near as easy as the early days when all that was necessary was to increase the clock speed of the CPU. It is for this reason that CPU designers pushed the single-threaded performance to the limits before attempting to adopt a multi-threaded approach to increase performance. When multi-threading is not enough, then, and only then, is cluster computing considered an option.

Many in the cryptocurrency industry have attempted to solve the scalability issue by immediately moving to a "cluster" solution without fully exploring what is technologically possible on a single core of a single computer.

2) LMAX Disruptor

The LMAX Disruptor provides a case study on an architecture with a high degree of scalability and performance, showing what is achievable within a single execution thread. LMAX is a retail trading platform that aims to be the fastest

exchange in the world. They have been generous enough to share what they have learned publicly.

A brief overview of LMAX architecture:
The Business Logic Processor is where all of the sequential transactions and order matching is processed. It is a single thread that is able to process millions of orders per second. This architecture is readily ported to the realm of cryptocurrencies and blockchain designs. The role of the Input Disruptor is to gather orders from users from many different sources and assign them a deterministic order. After assigning them an order they are replicated, logged, and broadcast to many redundant business logic processors.

The tasks of the Input Disruptor are parallel and easily farmed out to a cluster of computers. An Output Disruptor takes care of notifying anyone who cares about the results. This is also a parallel task. Ultimately, LMAX was able to process 6 million transactions per second through the Business Logic Processor using a single core of a commodity CPU using the Java virtual machine. If LMAX can achieve 6 million transactions per second, then certainly there is no need for cryptocurrency and smart contract platforms to reach for clustered solutions when they are not even processing 10 transactions per second.

To implement a high performance blockchain, Borderless must adopt the same techniques used by LMAX. Several key fundamentals must be met: Keep everything in memory, avoid synchronization primitives (locks, atomic operations) and minimize unnecessary computation in the business logic processor. Memory is becoming cheaper every day because it is extremely parallel in its design. The amount of information that is required to track the account balance and permissions of every person on the Internet is less than 1 Terabyte of RAM, which can be purchased for less than $15,000 and installed on commodity (high-end) server motherboards. Long before 3 billion people adopt the system, this kind of hardware will be in the average desktop. The real bottleneck is not the memory requirements, but the bandwidth requirements. At 1 million transactions per second and 256 bytes per transaction, the network would require 256 megabytes

per second (1 Gbit/sec). This kind of bandwidth is not widely available to the average desktop; however, this level of bandwidth is a fraction of the 100 Gbit/s that Internet 2 furnishes to more than 210 U.S. educational institutions, 70 corporations, and 45 non-profit and government agencies.

Therefore, blockchain technology can easily keep everything in RAM and scale to handle millions of transactions per second if it is designed properly.

3) Assign IDs To Avoid Hashes

In a single threaded system, CPU cycles are a scarce resource that need to be conserved. Traditional blockchain designs use cryptographic hashes to generate globally unique IDs that are statistically guaranteed to never have a collision. The problem with these hashes is that they require significantly more memory and more CPU cycles to manipulate. It takes significantly more CPU time to look up an account record by hash than with a direct array index. For example, 64 bit integers are easier to compare and manipulate than 160+bit IDs. Larger hash IDs means there is less room in the CPU cache and that more memory is required. On modern operating systems, infrequently accessed RAM is compressed, but hash identifiers are random data that is not compressible. Fortunately, blockchains give us a means to globally assign unique IDs that do not conflict with one another, so it is possible to completely remove the need to use hash-based identifiers like Bitcoin addresses to refer to an account, balance, or permission.

4) Remove Signature Verification From Business Logic Processor

All transactions on cryptocurrency networks depend upon cryptographic signatures to validate permissions. In the general case, the permissions required can change as a result of other transactions. This means that permissions need to be defined in terms that require no cryptographic calculations within the Business Logic Processor.

To do this, every public key needs to be assigned a unique and immutable ID. After an ID has been assigned, the Input Disruptors can verify that the signature provided matches the ID specified. By the time the transaction makes it to the

Business Logic Processor, the only remaining step is to check the IDs.

This same technique can be used to remove pre-condition checking on any immutable object with a static ID.

5) Design Transactions For Static Validation

Many transaction properties can be checked statically, without the need to reference the current global state. These checks include range checking of parameters, de-duplication of inputs, sort order of arrays, etc. Generally speaking, many checks can be performed if the transaction includes the data it "assumes" about the global state. After these checks are performed, all that is necessary for the Business Logic Processor to do is make sure the "assumptions" are still true, which can usually be boiled down to checking a modification timestamp on objects referenced relative to the time the transaction was signed.

6) Smart Contracts

Many blockchains are adopting a general purpose scripting language to define all operations. These designs end up defining the "Business Logic Processor" as a virtual machine and all transactions are defined as scripts to be run by the virtual machine. This approach takes the single-threaded limitations of a real CPU and compounds them by forcing everything through a virtual CPU. A virtual CPU, even with Just-In-Time compilation, will always be slower than a real CPU, but pure speed of calculation isn't the only issue with the "everything is a script" approach. When transactions are defined at such a low level, it means that most of the static checks and cryptographic operations get sucked back into the Business Logic Processing and the overall throughput falls. A scripting engine should never require a cryptographic signature check to be performed even if it is done through a native call.

Based upon the lessons we learn from LMAX, we know that a virtual machine for a blockchain should be designed with single-threaded performance in mind. This means it should be optimized for Just-In-Time compilation from the beginning, and that the most frequently used smart contracts should be supported natively

by the blockchain, leaving only the rarely-used custom contracts to run in a virtual machine. These custom contracts should be designed around performance, which means the Virtual Machine should limit the addressable memory to something that will fit within the CPU cache.

7) Objected Oriented Data Model

One of the benefits of keeping everything in memory is that the software can be designed to mimic the real-world relationships of data. This means that the Business Logic Processor can quickly follow in-memory pointers to the data it needs, rather than being forced to perform expensive database queries. It also means that data can be accessed without copying it, and that the data can be modified in-place. This single optimization offers an order-of-magnitude performance gain over using a database- based approach.

Borderless built a high-performance blockchain by removing all calculations that are not part of the critical, order-dependent, evaluation from the core business logic, and designing a protocol that facilitates these kinds of optimizations. This is what Borderless has done.

4. Functional Characteristics of Bitcoin Wonder System

1) Open Source and Complete Transparency

Bitcoin Wonder source code has been published in, the global third largest open source site, github.com. All communication is open sourced and is supported by a very open community. There is no other place can be as transparent as Bitcoin Wonder.

2) Privacy

By using Bitcoin Wonder you can protect your privacy. Like Bitcoin, all the transactions are completely open, without getting bonded to your true identity. No IRS documents will be required. Neither will anyone asks for the copy of your passport, driver's license, utilities and credit report.

3) Elliptic-curve cryptography

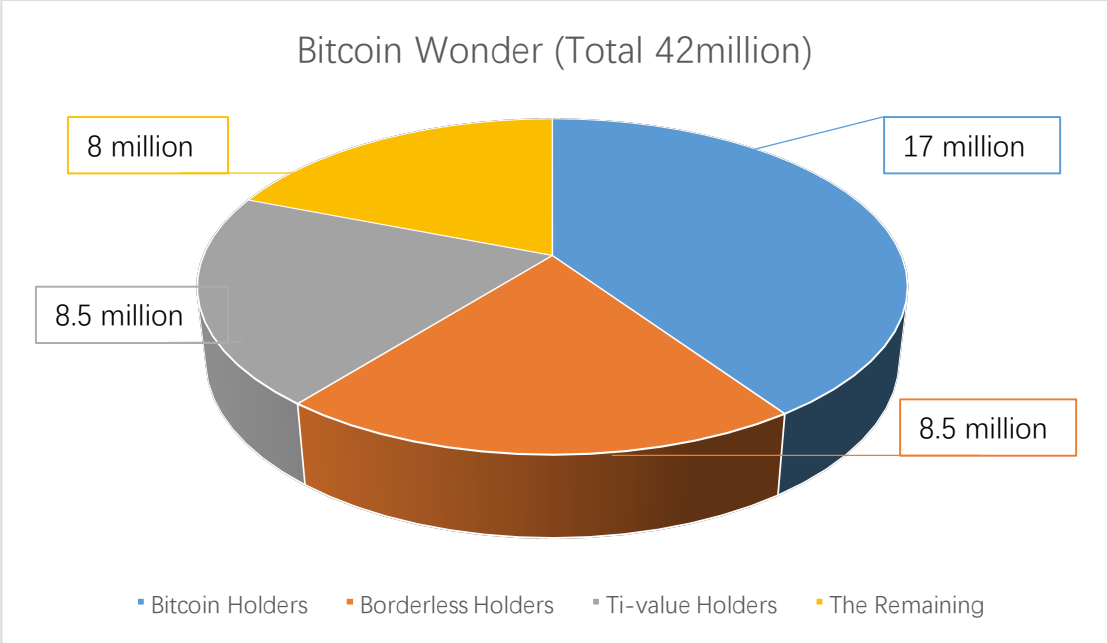
Elliptic-curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. The use of elliptic curves in cryptography was suggested independently by Neal Koblitz and Victor S. Miller in 1985.

```
   int secp256k1_ecdsa_verify(const secp256k1_context_t* ctx, const unsigned char
*msg32, const unsigned char *sig, int siglen, const unsigned char *pubkey, int
pubkeylen) {

   secp256k1_ge_t q;
secp256k1_ecdsa_sig_t s;
secp256k1_scalar_t m;
int ret = -3;
DEBUG_CHECK(ctx != NULL);
DEBUG_CHECK(secp256k1_ecmult_context_is_built(&ctx->ecmult_ctx)); DEBUG_CHECK(msg32 !=
NULL);

   DEBUG_CHECK(sig != NULL); DEBUG_CHECK(pubkey != NULL);

   secp256k1_scalar_set_b32(&m, msg32, NULL);

   if (secp256k1_eckey_pubkey_parse(&q, pubkey, pubkeylen)) { if
(secp256k1_ecdsa_sig_parse(&s, sig, siglen)) {

   if (secp256k1_ecdsa_sig_verify(&ctx->ecmult_ctx, &s, &q, &m)) { /* success is 1,
all other values are fail */
ret = 1;

   } else {

ret = 0; }

       } else {
          ret = -2;
```

```
}
} else {
ret = -1; }
return ret; }
```

5. Bitcoin Wonder Distribution

The total amount of BCW is 42 million, twice than that of Bitcoin. Among this total supply of 42 million, 17 million will be allocated to Bitcoin holders at a 1 BTC: 1 BCW ratio. 8.5 million will be allocated to Borderless holders (more details on the ratio allocation method further on) and another 8.5 million will be airdropped to Ti-value holders. The remaining 8 million will be kept by Bitcoin Wonder itself for promotion.



Bitcoin Wonder will be released on several established exchanges where Bitcoin users can get Bitcoin Wonder freely at a ratio of 1 BTC: 1 BCW. While the Borderless users will get Bitcoin Wonder according to the real-time value of their BDS which is calculated through Bitcoin. In another word, the more Borderless you have, the more Bitcoin Wonder you' ll get.