The Unity Protocol

A Manifesto for Digital Global Unity

By: Joshua (김대성) Tobkin & Jonathan Jones
Reviewed by Dr. Hermann Liu Ph.D. and Dr. Arcady Novosyolov Ph.D.
July, 2018 Seoul, Korea

Introduction:

"Great acts are made up of small deeds" - 老子 (Lao Tzu)

Imagine a world where blockchain participation is not limited to those that can afford specialized mining equipment or the ability to stake large amounts of currency. It would certainly be a more equitable, democratic ecosystem.

Blockchain technologies enable digital identity and ownership, and as a result, a voice. We believe these are fundamental human rights, that all humans should be allowed to securely control their identity, data, and finances without third-party intervention. Unity Protocol's goal is to usher in the next generation of human self-governance and democracy by enabling equal rights to participate in the blockchain ecosystem.

We have unique opportunities in front of us with the emergence of next generation consensus protocols. These protocols are an entirely new class of consensus frameworks for distributed systems that don't rely on wasteful Proof of Work (PoW) mining nor game-theoretical-behavioral-economic systems like Proof of Stake (PoS).

We believe that all elegant consensus protocols must have at their core a strong level of mechanical ingenuity. Satoshi Nakamoto's PoW achieves this. It is brilliant, especially for its time. Nonetheless, the centralization of mining pools and the wastefulness of computers singularly racing towards an arbitrary number causes severe externalities on our environment, and also ironically upon decentralization itself.

Proof of Stake systems, though much faster, are not only easily centralized, but also dependent upon humans always behaving rationally. As history has taught us, human rationality has limitations.

This is why we believe that an elegant consensus protocol must at its core have mechanical ingenuity (Hard Tech), whilst having the ability to add 2nd layer incentives such as Proof of Stake or Proof of Reputation (Soft Tech), *appropriate only as an additive layer to further increase network security*.

The Unity Protocol achieves these objectives without the wastefulness of PoW or the plutocracy of PoS systems.

Conceptualize a world where blockchain participation is lightweight enough to run from your mobile phone, and yet is still as secure as a truly decentralized PoW network. Imagine a dynamic system that scales better with greater speed and security when more validators join the network. This is what Unity Protocol aims to achieve. It is our hope that over the following pages we will be able to elucidate how the next generation class of distributed consensus technologies will permeate society.

Description and definitions of Components:

API: a set of functions and procedures that allow the creation of applications which access the features or data of an operating system, application, or other service.

Artificial Intelligence: the term "artificial intelligence" is applied when a machine mimics "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving".

Blockchain: a digital ledger in which transactions made are secure are recorded chronologically publicly.

Butterfly Effect: In chaos theory, the butterfly effect is the sensitive dependence on initial conditions in which a small change in one state of a deterministic nonlinear system can result in large differences in a later state.

Chaos Theory: Chaos theory is a branch of mathematics focusing on the behavior of dynamical systems that are highly sensitive to initial conditions

Cryptographically Signed: A signing algorithm that, given a message and a private key, produces a signature. A signature verifying algorithm that, given the message, public key and signature, either accepts or rejects the message's claim to authenticity

DAO: A decentralized autonomous organization (DAO) is an organization that is run through rules encoded as computer programs called smart contracts.

DHTs: A distributed hash table (DHT) is a class of a decentralized distributed system that provides a lookup service similar to a hash table: (key, value) pairs are stored in a DHT, and any participating node can efficiently retrieve the value associated with a given key.

Dimensional Scaling: Ability to scale horizontally and vertically as needed in order to maintain the security, throughput, and integrity of the network.

Epochs: a particular period of time marked by distinctive features, events, etc.

Genesis Algorithm: The algorithm and Random Number Generator to sets up the early conditions of the network.

Hard Tech: Technologies like hardware and software that are codified or mechanical in nature.

Hashing Algorithm: A hash function is any function that can be used to map data of arbitrary size to data of a fixed size. The values returned by a hash function are called hash values, hash codes, digests, or simply hashes. Hash functions are often used in combination with a hash table, a common data structure used in computer software for rapid data lookup

Initiation Algorithm: Initiates the actual selection of Oracles, node assignment to tribes, and other relevant steps in order to kickstart the network.

Ledger: A collection of transactions

Machine Learning: Machine learning is a subset of artificial intelligence in the field of computer science that often uses statistical techniques to give computers the ability to "learn" (i.e., progressively improve performance on a specific task) with data, without being explicitly programmed.

Master Algorithm: In our system a Master Algorithm produces a Master Count, and is designed to output a large random number.

Master Count: The output of the Master Algorithm.

Nodes: Computers connected to the network

Oracles: An oracle, in the context of blockchains and smart contracts, is an agent that finds and verifies real-world occurrences and submits this information to a blockchain to be used by smart contracts.

Power Law Distribution: a power law is a functional relationship between two quantities, where a relative change in one quantity results in a proportional relative change in the other quantity, independent of the initial size of those quantities

Random Number Generators (RNGs): Randomization Algorithms

Smart Contracts: A smart contract is a computer protocol intended to digitally facilitate, verify, or enforce the negotiation or performance of a contract. Smart contracts allow the performance of credible transactions without third parties

Soft Tech: Social technologies that are conceptual and not based on physical criterion. Examples of Soft Tech include social norms and behavioral economic game theory.

Tribes: Quorum or cluster of Nodes

Tribal Leaders: The node selected to handle the first level of validation

Unity Event: When there is a majority consensus on the validity of a transaction

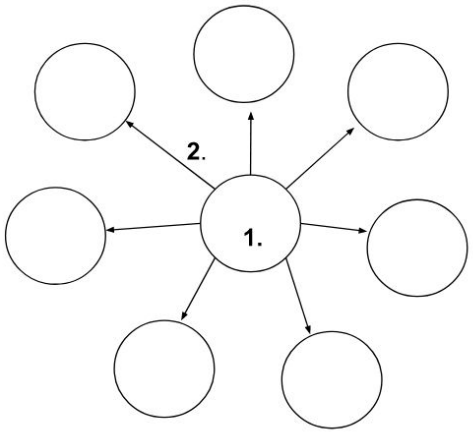
Unity Pillars: Turing complete Distributed Hash Tables

Validator: a node that checks the blockchain ledger to determine if the transaction to be processed is valid

How Does The Unity Protocol Work?

It starts with the user, or a group of users transacting with each other (1), whose transaction is broadcast to 7 Decentralized Oracles (2).

UnityChain.net

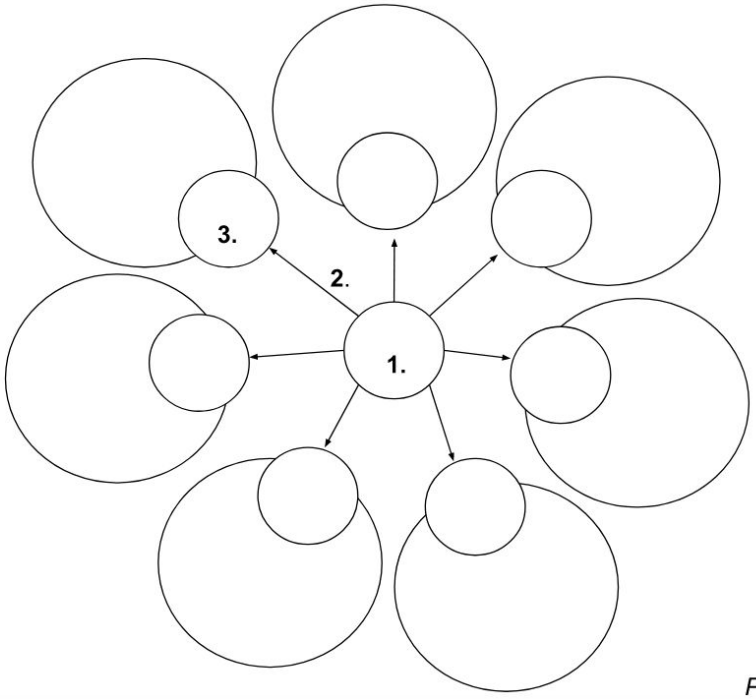


1. Users Transact
2. Broadcast to 7 Oracles

Please keep confidential

Nodes are evenly distributed into tribes (clusters of nodes), with each tribe associated with one Oracle (3).

UnityChain.net



3. Oracles Connected to 1 Tribe Each

Please keep confidential

In our system, an true random number generator (RNG) is paramount. Most RNGs are “pseudo-random” because they are generated from a closed-circuit system like a computer. Over time macro patterns emerge, hence the nomenclature of “pseudo-random.”

We introduce Chaos Theory and the Butterfly Effect into our system to generate an authentic random number (aRN). We source data inputs that are beyond the closed circuits of computer systems, such as geothermal data, weather patterns, global temperatures, and more. When appropriate, we may use large, static datasets such as past stock prices to add more entropy to our system.

The key is to use external data sources that have high levels of unpredictability, but may also be cross verified easily, and then to use those out-of-circuit values inside the RNG.

Expanding upon this, in this next step the 7 Oracles cycle through available external data sources and agree upon one. For the purposes of this example, let's suppose that the “Current Temperatures of the World” data source was selected. The Oracles then cycle through all 193 countries in the world and each individually select a country (each one unique).

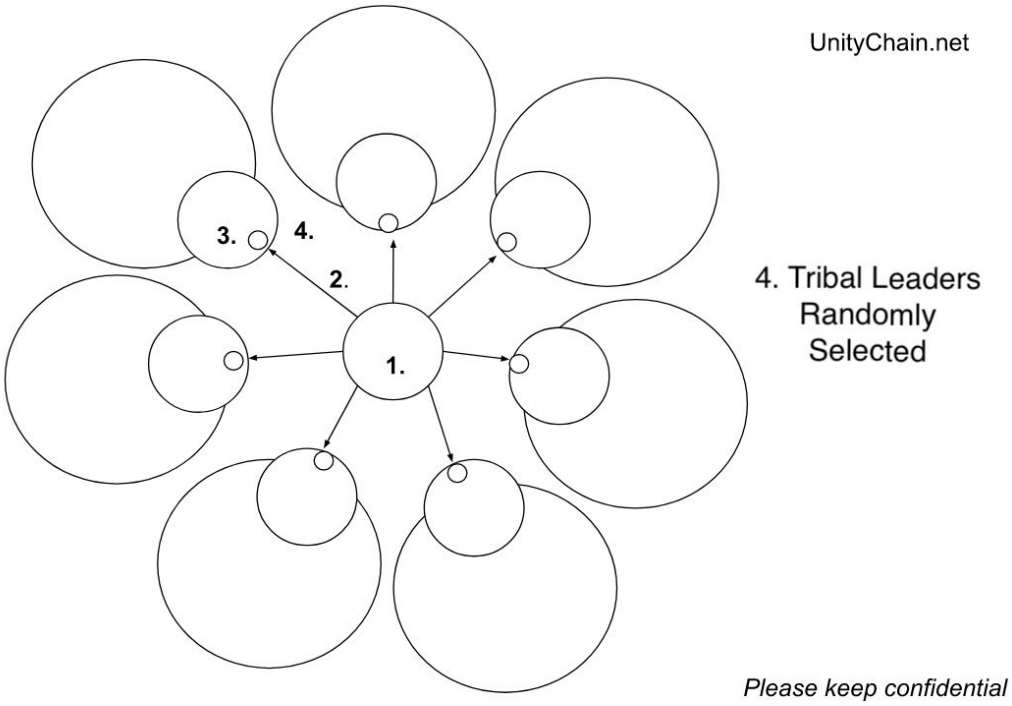
Next they individually select a city within their selected unique country and look up its current temperature through trusted APIs (generally provided by weather stations or governments of the world). The Oracles finally check each other’s temperature values to ensure accuracy. In our system we’ll only use absolute values; this way readings from Iceland will always be positive integers, even during the middle of winter.

At this stage, we end up with 7 numbers derived from 7 cities in the world’s verified temperature. These numbers are then placed into the “Master Algorithm,” which is in its simplest form an equation that has 7 variables that are replaced with the values from the prior step. These Master Algorithms are designed to always generate a large, authentic random number (aRN).

This generated number is called the “Master Count” (MC). Oracles use the Master Count to cycle through each node within their respective tribes until the MC is reached.

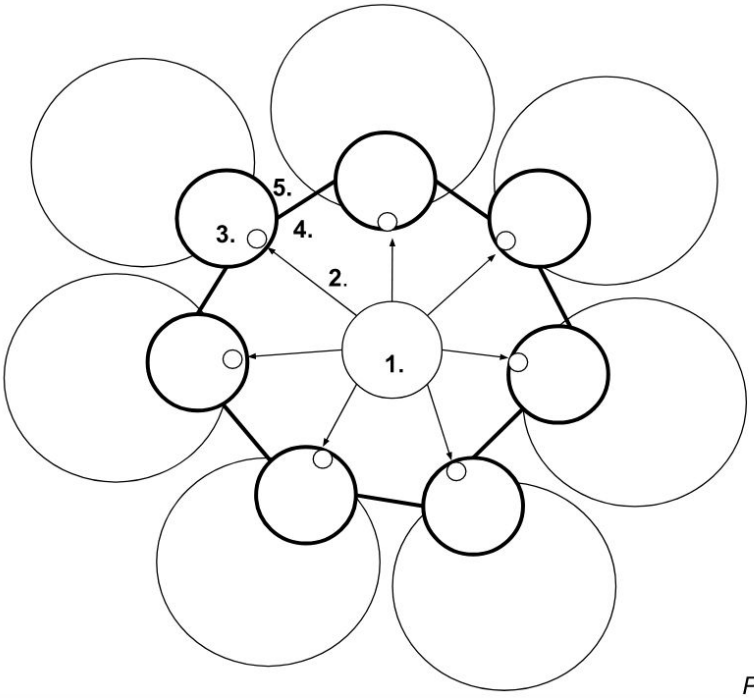
Simple Example of a Master Algorithm:
[Largest #] * [2nd Largest #] * [3rd Largest #] * [4th Largest #] * [5th Largest #] * [6th Largest #] * Pi **DIVIDED BY** [Smallest #] = **Master Count (MC)**

Each Oracle next sends a broadcast to their individual tribes to get a roll call of respondent nodes and takes a snapshot of their respective tribes. Oracles apply the MC to the roll call list until the MC is reached. The node that is selected is appointed “Tribal Leader,” or lead validator for the next 5 minute cycle. This is how we achieve 7 randomly selected tribal leaders (4).

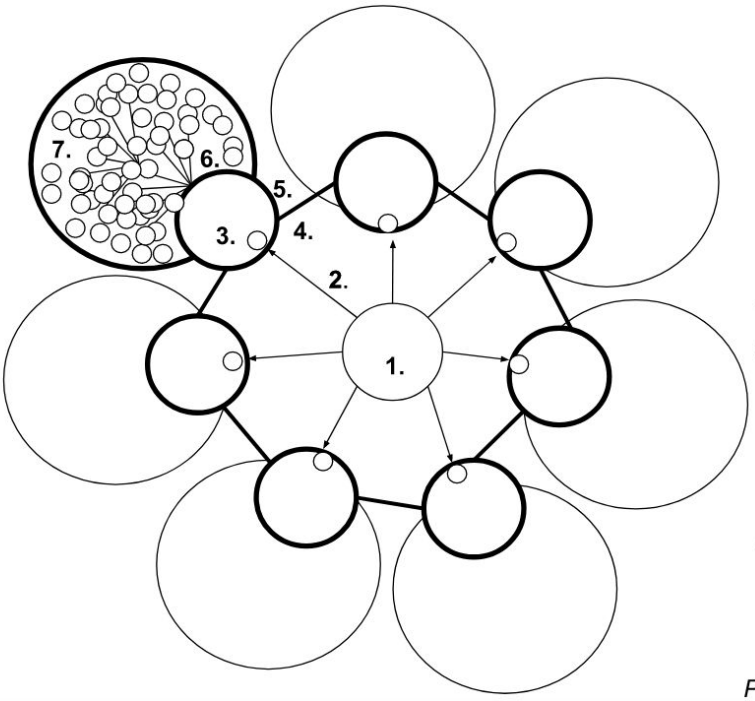


Next, the selected tribal leaders each perform validation on the original transaction from step (1) by checking the distributed ledger to see if the transacting accounts have the right balances or the correct access rights. The tribal leaders check the ledger and then broadcast their answers to each other. "Yes" means that the transaction is valid, and "No" means that it is not.

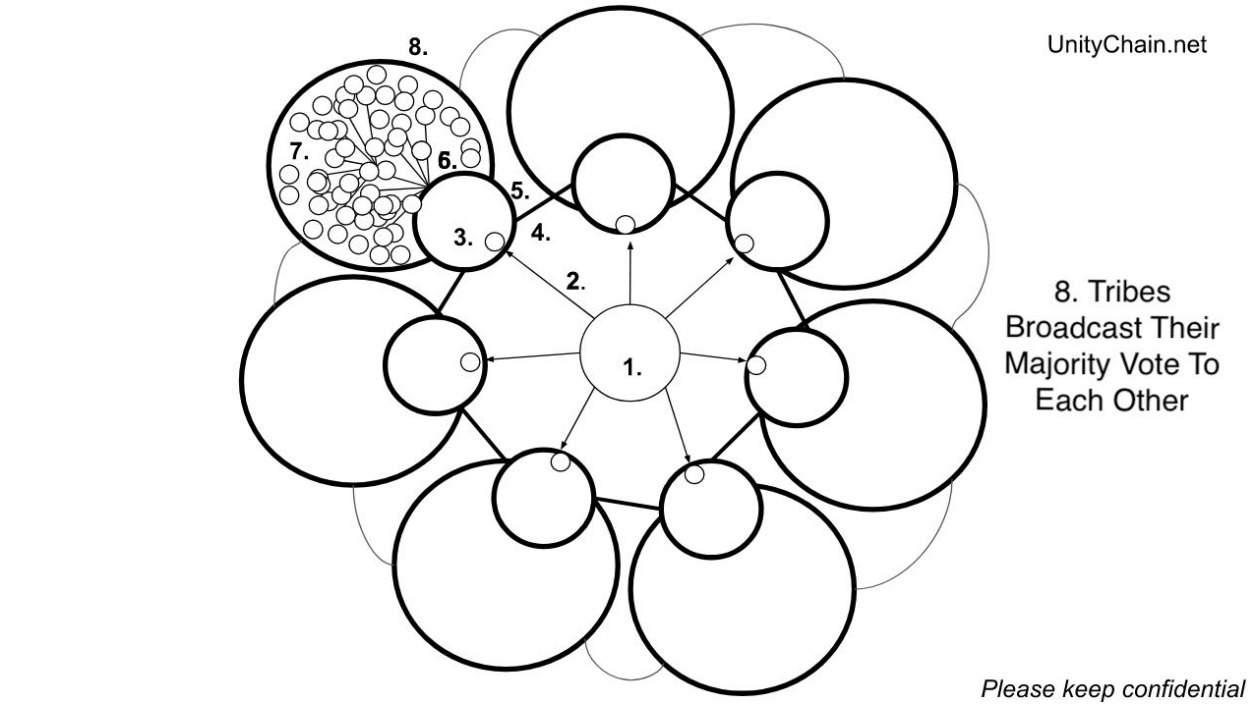
If they all agree, this is a “**First-Degree Unity Event,”** the first level of consensus among tribal leader nodes, who otherwise would not know each other (5). If the tribal leaders are unable to reach consensus on this level, the transaction may be dropped and unable to propagate to the rest of the network. In other words, all tribal leaders *must* come to an agreement for the transaction to continue through its validation process.



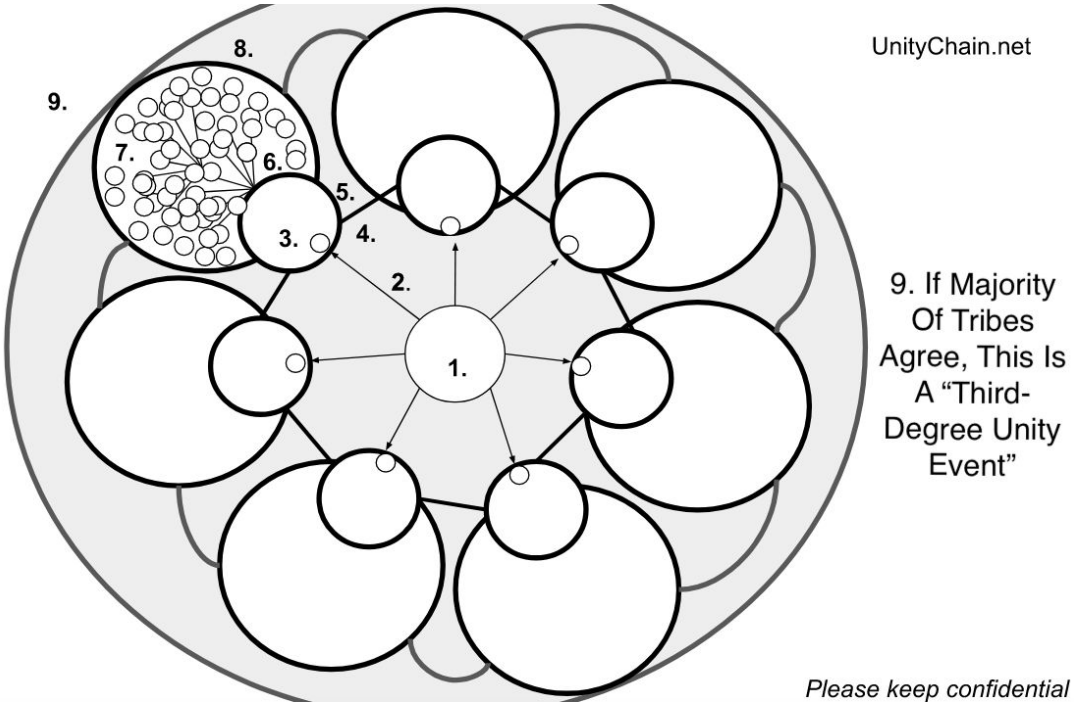
If all the tribal leaders are in agreement, the transaction is broadcast by the tribal leader to their respective tribes, with each tribe member passing along the message to each other until a majority of that particular tribe has seen the broadcast and validated the transaction (6). If a majority of that particular tribe comes to an agreement, this is a "**Second-Degree Unity Event,"** a consensus among **intra**-tribal nodes (7).



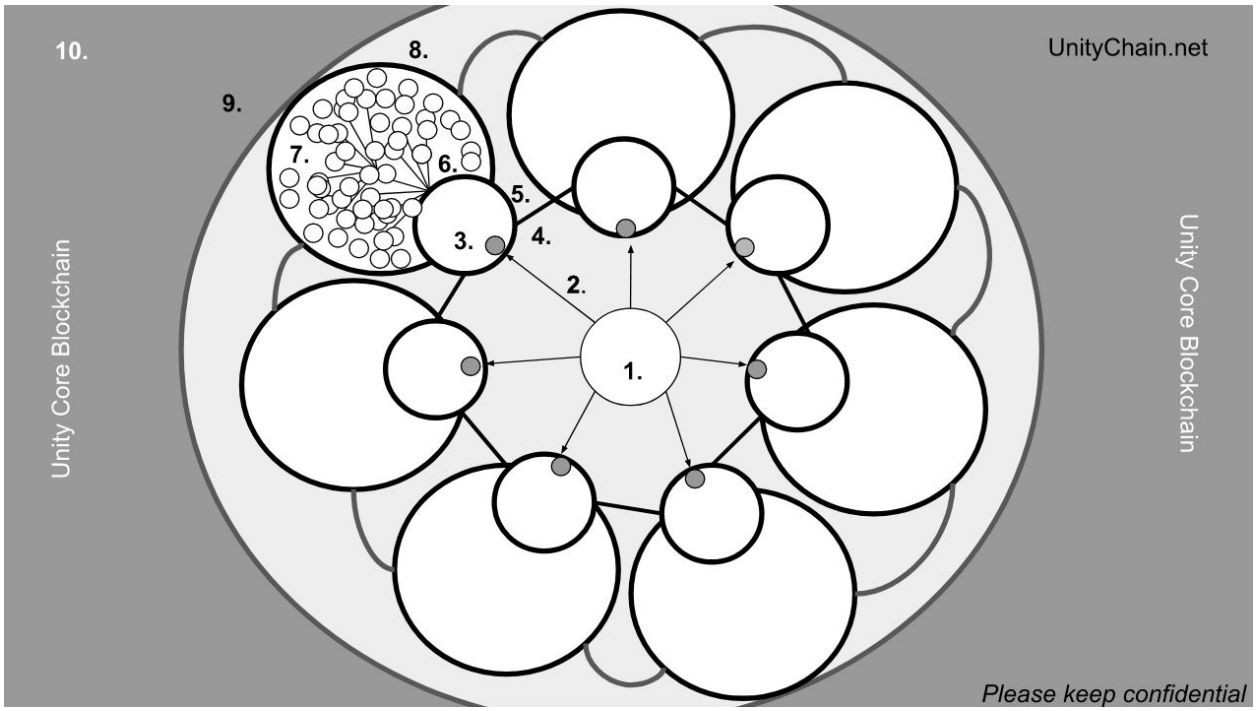
The tribes, as collective groups, next broadcast their majority vote on the transaction's validity to each other and seek consensus (**inter**-tribe consensus) (8).



If a majority of all the tribes reach an agreement, **inter**-tribal consensus, this is a "**Third-Degree Unity Event,"** the most secure and trusted form of transaction validation (9).



At this last step, if a majority of the tribes are in consensus, the transaction is finally cryptographically signed by the tribal leaders and allowed to be recorded to the distributed ledger or blockchain (10).



This last step of the system is called ***Proof of DTS***, “Proof of Digital Tribunal Signature.”

At any of these stages if determined consensus levels are not met, the transaction will not be allowed to be recorded to the ledger.

Tribal Leaders in our system are leaders for only 5 minutes at a time *(this number can be titrated as necessary for security)*. Also considering the tribal leader selection process can take up to 15-20 seconds, the Oracles will begin preparing the next aRN to be used in the next Tribal Leader selection process approximately 1 minute before the end of the existing cycle. This process may continue in an infinite loop.

Why Does It Work? (The Math)

In a system that has 10,000 nodes evenly distributed into 7 tribes, there are 1,428 nodes per tribe available to be selected as Tribal Leader. The probability of being selected as a tribal leader is $1/1428$.

Denote $N = 10000$ the number of nodes, $t = 7$ the number of tribes, and $n = N/t = 1428$ the number of nodes per tribe.
t

The probability that any single leader will be correctly predicted equals $p_l = 1/n = 1/1428 = 0.0007$. The probability that ALL t leaders will be correctly predicted equals to $p_l^t = 8.26 \cdot 10^{-23}$.

$8.26 \cdot 10^{-23}$ is the same number as 0.00000000000000000000082585261.

That’s the probability of being able to predict exactly which 7 Tribal Leaders will be selected in each successive round.

Another way to look at this:
On average, you can only predict 1 out of 12,108,698,122,149,000,000,000 cycles accurately which tribal leaders will be selected.[1]

To read that number semantically, it is:

[1] 1 divided by 0.00000000000000000000082585261 = 12,108,698,122,149,000,000,000

twelve sextillion, one hundred eight quintillion, six hundred ninety-eight quadrillion, one hundred twenty-two trillion, one hundred forty-nine billion cycles on average before being able to predict which exact 7 Tribal Leaders will be selected.

To help you understand this number in terms of scale, your chances of predicting who the 7 Tribal Leaders will be accurately is akin to selecting the correct *single grain* of sand out of all the sand grains in the entire world!

Actually, it turns out you are 1,614 TIMES *more likely* to guess the exact grain of sand[2] than you are able to predict the 7 Tribal Leaders!
12,108,698,122,149,000,000,000/(7.5 x 10^18) = 1614.5

Needless to say, the probability of predicting the exact outcome of Tribal Leaders is infinitesimally small.

For intermediate values of correctly predicted leaders one would use the binomial distribution with parameters $t,\ p_l$. Thus the probability that exactly k out of t leaders will be predicted correctly equals

$$p(k) = C_t^k p_l^k (1 - p_l)^{t-k},\ k = 0,\ 1,\ ...,\ t.$$

Here C_t^k stands for the number of combinations from t by k.

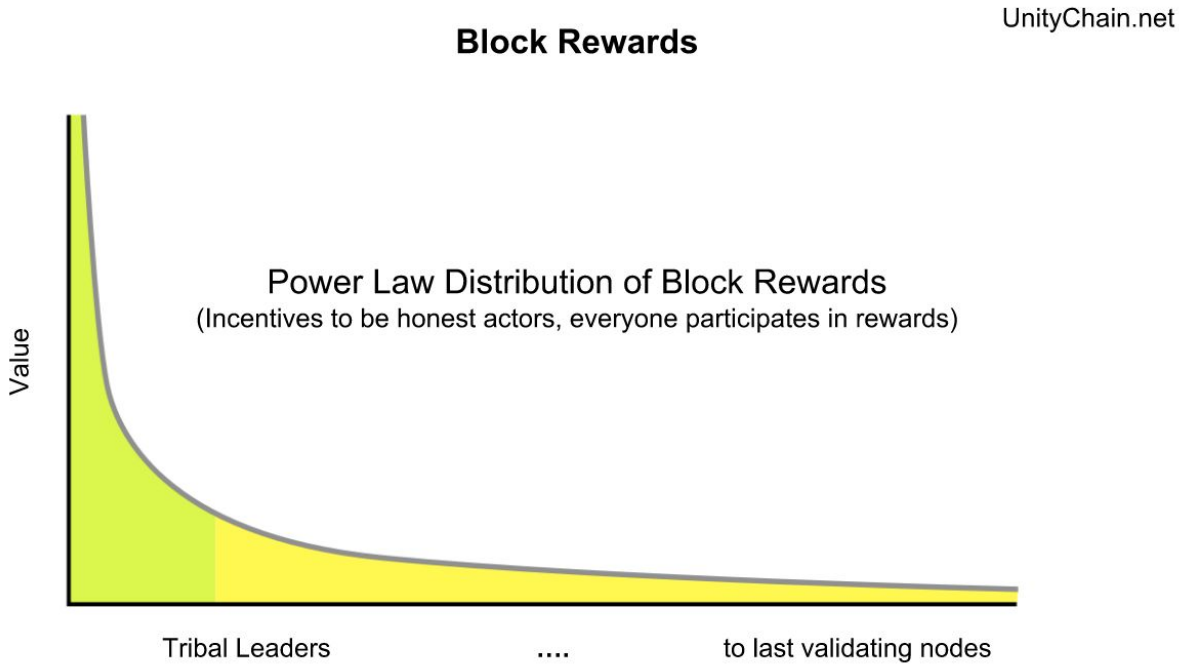
Since in our system Tribal Leaders may be elected for 5 minute terms, on average it would take 115,189,289,594,264,000 years[3] to correctly predict the next set of tribal leaders.That's one hundred fifteen quadrillion, one hundred eighty-nine trillion, two hundred eighty-nine billion, five hundred ninety-four million, two hundred sixty-four thousand years.

Power Law Economics (Block Rewards):

Every validator in our system receives block rewards for participating in consensus. By disproportionately rewarding positive behavior, incentives are aligned. Our network is self-incentivized: when more nodes are required to help transaction load, block rewards dynamically increase to encourage more distributed participation to meet the network's demand.

---

[2] *(7.5×10^{18}) grains of sand - David Blatner. "Chapter 1 numbers." *Title: Spectrums*, 1.ed,London: Bloomsbury Publishing Plc, 2012.
[3] 12,108,698,122,149,000,000,000 cycles * 5 mins / 60 mins / 24hrs /365 = years



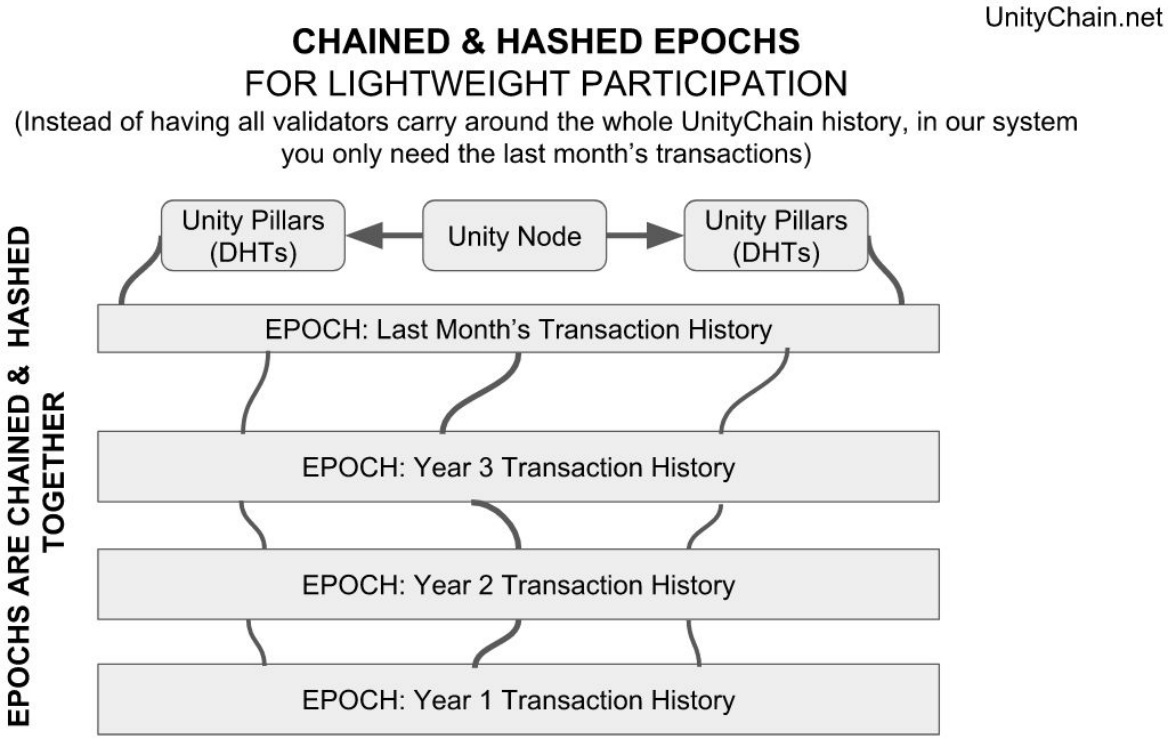
More information about our Token Economics will be provided in the near future.

Chained and Hashed Epochs for Lightweight Participation:

Instead of having all validators carry around the whole UnityChain history, in the Unity Protocol you only need to last month's transactions.

This is achieved by chaining Epoch's, which are periods of time (usually months or years), together through hashing algorithms. This data will be stored on Unity Pillars, which are decentralized Distributed Hash Tables. Nodes simply reference Unity Pillars when validating transactions as a secure reference point. Unity Pillars will synchronize with each other often. Some nodes may carry around the entire blockchain history as a full node. Though not necessary, there will be benefits and system rewards to incentivize those that help secure the decentralization of the network.

UnityChain.net



Sybil-Attack Prevention:

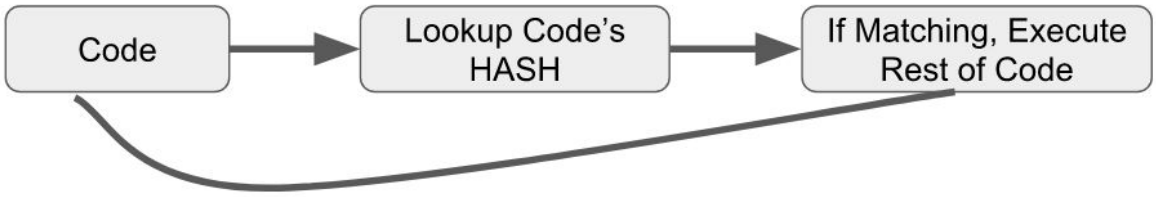
1. KYC/KYM and Proof of Participation (PoP):

In order to prevent sybil attacks, we will use zero-knowledge proofs (ZKPs) as a means of achieving Know Your Customer (KYC) and Know Your Machine (KYM).

ZKPs, along with zkSNARKS, will enable us to provide digital identities to individuals that want to run validating machines in our network, without requiring them to disclose their full legal identities. KYC and KYM is not necessary to have an account address in our network. It is only necessary if you want to validate and participate in block rewards.

Our system will keep an immutable log of all the votes that each machine has cast, which is a proof of participation (PoP). This is useful when identifying machines that are trying to game the system by voting “yes” when the consensus was “no”, and vice versa. Those devices that vote contrary to the majority vote beyond an acceptable threshold may be suspended or even banned from the system.

2. Tamper-Proof Protocol (Self-Referencing Programming):



Code in our system will always require an initial formal proof upon joining the network. Any system upgrade or code that is approved in our decentralized governing body is hashed and stored on a "Unity Pillar," which is a turing complete distributed hash table (DHT).

Unity Pillars run the hashing algorithm on the node's original code and then compare its hash to the approved code's hash. If the hashes match, we know the code is what it is expected to be. Unity Pillars then send a signal back to the node, allowing it to complete the rest of its coded execution. Code cannot complete execution unless receiving this signal from the protocol.

If a node is caught running an unapproved version of the protocol, it may be banned or severely punished.

3. Auto-Scaling Network Self-Preservation Powered By Artificial Intelligence and Machine Learning

Our system can auto-scale tribes, oracles, and tribal leaders when it detects abnormal behavior such as nodes or transactions failing beyond the expected standard deviation. Although this will affect the speed of the network, we optimize for security and will only scale back down when the system detects that the threat has been expelled.

Auto-Scaling to 11 Tribes/Tribal Leaders/Oracles

In subsequent cycles, the system can decide to auto-scale to 11 Tribes/Tribal Leaders/Oracles, which exponentially makes spoofing a transaction more difficult. All first-degree Unity Events *require all tribal leaders to be in agreement* before the transaction is able to propagate through the network.

Assuming we have 10,000 nodes evenly distributed into 11 Tribes, we now have 909 nodes per tribe available to be selected as Tribal Leader.

The probability of being selected a Tribal Leader equals:

$t = 11,\ n = 909$, so $p_l = 1/n = 1/909 = 0.0011$

The probability of predicting all 11 tribal leaders equals: $2.86 \cdot 10^{-33}$

Another way to express this is that on average it would take 350,108,548,905,788,000,000,000,000,000,000 cycles[4] before correctly predicting the correct exact 11 Tribal Nodes.

Auto-Scaling to 13 Tribes/Tribal Leaders/Oracles

If necessary, our system can even auto-scale to 13 (or more) Tribes/Tribal Leaders/Oracles, which exponentially decreases the probability of predicting the Tribal Leaders.

Assuming we have 10,000 nodes evenly distributed into 13 Tribes, we now have 769 nodes per tribe available to be selected as Tribal Leader.

The probability of being selected a Tribal Leader equals:

$t = 13,\ n = 769$, so $p_l = 1/n = 1/769 = 0.0013$

The probability of predicting all 13 tribal leaders equals: $3.04 \cdot 10^{-38}$

Another way to express this is that on average it would take 32,888,375,128,986,800,000,000,000,000,000,000,000 cycles[5] before accurately predicting the correct exact 13 Tribal Nodes.

[4] 1/2.85625701836002E-33
[5] 1/3.04058803780376E-38 cycles

With 5 minute Tribal Leader terms, it will take on average 312,865,060,207,257,000,000,000,000,000,000 years[6] before guessing the correct set of tribal leaders for the upcoming cycle.

4. Geolocation

We require that individual accounts cannot come from IP addresses of more than a specific allocation per general geolocation. Current active nodes with the same IP address will also be limited to join the network to mitigate harmful collusion.

For example, New York City has a total of about 15,000 IP addresses out of a population of 8.54 million people which is a ratio of 1/570. The number of similar IP addresses allowed to run a node at any given moment dynamically changes based on the total number of existing nodes on the network. If 10,000 nodes are running on the network we may only allow a maximum of 10 machines, for example, on the same IP address. *This requirement will not be fixed, but may be a metric the protocol considers when allowing new nodes to join the network.*

Statistically Significant Security Thresholds

So far we have only discussed the probability of being able to accurately predict the tribal leaders and we've discovered that it is, for all intents and purposes, virtually unpredictable in a system of over 10,000 nodes. But what happens when 1/3 of the network is compromised, even with tamper-proof protocols in place?

Suppose that 1/3 of nodes in each tribe are faulty. Moreover, consider a more general setting right now: the portion of faulty nodes in each tribe equals $q \in [0, 1]$. Then the probability that all tribal leaders are faulty equals $p(all) = q^t$. With $q = 1/3,\ t = 7$ we have $p(all) = (1/3)^7 = 0.000457$.

In a network of 10,000 nodes with 1/3 compromised, it will only take on average 2188 cycles before the 7 Tribal Leaders are all bad actors. However, this does not mean that false transactions will be able to propagate through the entire network and be inappropriately recorded in the blockchain. In our system, Tribal Leaders are still held accountable to their respective tribe members (**intra**-tribal consensus).

[6] 32,888,375,128,986,800,000,000,000,000,000,000,000*5min tribal leaders/60 mins/24 hrs in a day/365 days in a year

In this example, all 7 of the Tribal Leaders are by chance bad actors. That simply means the transaction will proceed to its next level of validation (from Tribal Leaders to **Intra**-tribal consensus), not that the transaction is added to the ledger.

In this next step, the tribal leader broadcasts the transaction to their respective tribes, and the tribe members independently validate the transaction. If the majority of the tribe does not agree that this is a valid transaction, the tribal leader's vote will be discarded. Also, the tribal leader will be demoted and replaced, or even potentially removed from the network.

This means that even if 1/3 of the network is compromised, the faulty transaction is likely to be caught and disregarded during the **intra**-tribal phase of consensus.
We stipulate that over 51% of the network must be compromised before bad transactions can penetrate the intra-tribal level consensus and create a false second-degree Unity Event.

Even still, this does not mean that a 51% attack on our system will be able to fully override our network completely for the following reasons:

- Even with 51% of the network compromised, all randomly-elected Tribal Leaders must come to a consensus. If even only 1 out of the 7 elected tribal leaders are good actors, a first-degree Unity Event is not achievable and the transaction will fail.

  In fact, it will still take 111 cycles on average before all 7 selected tribal leaders all uniformly are bad actors even when 51% of the network is compromised. And then still the transaction has to go beyond intra-tribal (Second-degree Unity Event) and inter-tribal (Third-degree Unity Event) consensus.

- The system can auto-scale when sybil threats are detected. When 11 and 13 Tribal Leaders are required, it will take on average 1,647 and 6,333 cycles respectively before all randomly elected tribal leaders are uniformly bad actors.

  Amidst a 51% attack, assuming tribal leader requirements have auto-scaled to 13 tribal leaders, it would still take an estimated 5 to 7 days minimum for an invalid transaction to reach the blockchain, which is plenty of time for the system to cross-check the network and dispel bad actors. In short, we believe sybil attacks are improbable and that if one does occur, our network will be resilient enough to

overcome it.

- In the event that the network is below 10,000 validators, the system may simply require more tribal leaders for the added security benefits.

Governance

(Double-Loop Liquid Democracy via Weighted Points Systems)

Blockchains are meant to be *everlasting,* immutable records. However, technology evolves at an unintuitive pace. Indeed, it is impossible to predict what technologies will be available in 10 years, and in 100 years even less so. As such, it is important that blockchains have a decentralized governance structure that allows for smooth system upgrades, unlike some networks today where slow decisions result in developmental lag.

UnityChain intends to assimilate all blockchain best practices as they arise. This will ensure our network is always relevant even deep into the future. In order to achieve a decentralized parliamentary system that is controlled by the network, we will introduce a weighted point system to initiate system upgrades.

With a weighted point system, a certain amount of points from the community must be accumulated before a certain protocol update is possible.

The best way to explain this is through a simple example:

Suppose a system upgrade requires 65% of all votes, with each participant in the network assigned 1 point. In a network of 10,000 nodes, it would require 6,500 points in order for the system upgrade to be achieved. This type of general weighted point system would allow for the network to vote on certain updates. Some upgrades may require even more of the network to vote in order to be pushed through.

Though this example is simplistic, the general concept of weighted point threshold requirements for certain protocol-level upgrades is important in order to ensure that the network is owned and governed by the community. The protocol, which is objective, will simply accept and implement whatever the network majority votes upon.

Furthermore, our governing system would allow participants to auto-assign their vote to a particular delegate whom they feel is aligned with their personal values. This

addresses voter fatigue, the idea that participants in a network do not want to vote every time. In the event a participant no longer feels that particular delegate represents his or her values, casting their vote directly or assigning it to a new delegate will be a matter of a few clicks. The ability to switch your vote with ease is a trademark quality of a direct liquid democracy.

Blockchain Interoperability

Blockchain interoperability and transaction unity will be managed through Smart Contracts suites, our internal decentralized exchange (UnityDEX), APIs, and SDKs.

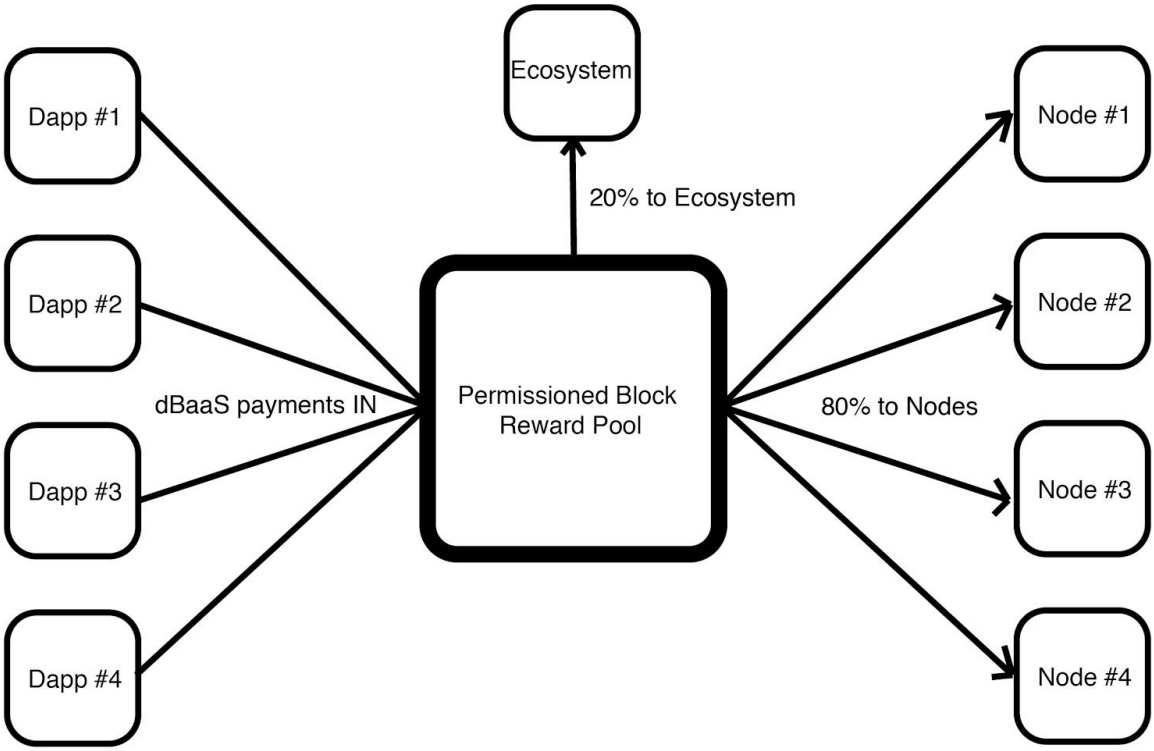
Business Models:

Random Access Protocol (RAP):
For a blockchain to be self-sustaining, it must have its own internal economic engines. One of our very first products will be our Random Access Protocol (RAP), which is pay-for-access to our authentic Random Number Generator. This tool will provide gaming sites and other computer systems aRNs (authentic random numbers) to help power fairplay. This service can be available through APIs and may cost our internal currency “Units” in order to operate.

As we regard highly the importance of authentic random numbers, we will create “The Foundation for Entropy,” which will be a non-profit R&D arm of UnityChain, with the mission to constantly improve our RNG techniques. We will keep a log of how each random number was created, which will be its “Proof of Randomness.”

Decentralized Blockchains as a Service (dBaaS):
We will also provide modular and configurable decentralized blockchains as a service (dBaaS) to small and medium enterprises. These will be permissioned blockchains, or side chains, that have configurable settings. Our community can be empaneled to participate in permissioned settings as validators for a negotiated price. Perhaps 80-97% of these fees will be distributed back to the network of objective participating validators. However, a small fee may be collected by the protocol for ongoing self-sustainability.

Dapps can pay a monthly fee to participate on our network with public or permissioned settings. That payment to the protocol goes directly to the Node Block Reward Pool, which gets distributed to participating nodes as block rewards.



Ecosystem Building:

UnityChain Ecosystem Pool is paramount to the future success of the network by creating incentives for Dapps to be built on the network. For example, new Dapps will be able to run on the network for free for a certain amount of time (6-12 months). Other incentives include development promotions through our grants.

Foundation Tokens:
Finally, we will establish the Unity Foundation, which will be allocated a certain amount of Units in order to maintain the decentralized sovereignty of the network.

Roadmap

Q1 2018: UnityChain's architecture, core concepts conceived

Q2 2018: Team formation, Light Paper validated and published, initial seed received, fundraising USA and Asia initiated. *(We Are Here)*

Q3-4 2018: Private Presale complete, Public Sale Announced, Test Net development and deployment.

Q1 2019: Public Sale complete. Partnership Ecosystem cultivation Unity Profiles (Identity/Governance), SDKs, and APIs made available. Community Growth Stage. Token Listed on exchanges.

Q2-3 2019: Mainnet Launch

Investment

We are currently a registered Delaware C Corp (Unity Chain Inc.). We are considering offers from strategic investors. To learn more please contact Hermann@UnityChain.net. We are currently raising a private pre-sale round for equity and tokens.

In Summary:

We believe the expected transaction throughput of our initial network can be around 10,000 transactions per second (tps), and with 7-10 seconds finality. Over time and with enough network participants, we believe this architecture will actually sustain more transactions more securely than even centralized systems like Visa/MasterCard.

Ultimately, our intention is to create a DAO, a Decentralized Autonomous Organization, that exists for the betterment of the entire species. While this is our goal, we know that it would be premature to assume we could build the perfect DAO in a short period of time. As a result, it is our intention to release the DAO fully within 21 years.

Until then, the network will be democratically supervised by a liquid governance structure that has the transparent intention to generate as much value for all stakeholders of the human race.

Is a KYM Network like this quantum resistant? What are some other attributes a protocol like this needs in order to be everlasting? These and more are some of the questions we're peering into on a daily basis.

If you're inspired and interested to join the team, please feel free to email us at Jillian@UnityChain.net

If you're an impact investor and you'd like to participate in our early equity round, please email Hermann@UnityChain.net

Remember...

"Great acts are made up of small deeds" - 老子 (Lao Tzu)

UNITY CHAIN