# Blockchain Infrastructure for Measuring Domain Specific Reputation in Autonomous Decentralized and Anonymous Systems

Craig Calcaterra, Wulf A. Kaal, Vlad Andrei

## Abstract

Autonomous decentralized validation of domain-specific reputation is a core decentalized infrastructure necessity.

The main design goals for measuring domain-specific reputation in autonomous decentralized and anonymous systems are to:

1) evaluate domain specific reputation across a wide variety of expertises

2) insure security, including resistance to Sybil attacks, tyranny of the majority, and 51% attacks

3) provide a minimal, robust core architecture which supports modular constructions for a wide range of use cases

The key features are:

1) a positive feedback loop which promotes productive cooperation between public and expert users

2) maximal freedom for experts in the creation of reputation verification protocols

3) complete autonomy—the economic structure is closed to off-domain influence; all power and fees are shared entirely by expert users, weighted by reputation

4) meritocracy—the platform supports anonymity, rewarding ability and productive contributions over personal identity

The creation of a transparent, fair, and predictable system that facilitates certainty of outcomes, security, and efficiency benefits society with the advancement of justice and prevention of corruption on a global scale.

## Table of Contents

# 1 Motivation

Public blockchains are decentralized, autonomous systems that allow transactions between anonymous parties. In particular, the Ethereum blockchain creates a Turing-complete transaction system via smart contracts. Smart contracts are computer programs that run on top of the Ethereum infrastructure, often for the purpose of executing transfers of value (e.g., currency, property, reputation, work). These transactions cannot be altered and are run without intermediaries, which leads to much lower marginal transaction costs (cents to dollars), regardless of the amount involved. These smart contract blockchain transactions follow a code-is-law paradigm, meaning they are self-regulating and self-enforcing, which reduces the need for traditional legal enforcement mechanisms as well as other traditional business execution mechanisms. Smart contracts promise to be increasingly important in the global economy, for clarifying intent, creating transparency, and removing the inefficiencies of intermediators, guarantors, and regulators.

However, in any system with anonymous actors, there is a fundamental problem of establishing trust between parties before business can confidently proceed. Traditionally, trust is established through centralized social structures, between parties with well-established identities. There are two requirements in order for the crypto economy to continue to develop while avoiding the inefficiencies of centralized regulating authorities: 1) a system for evaluating reputation/trust; 2) a fair dispute resolution system which guarantees certainty of outcomes. To successfully serve the crypto economy, any such system must also be decentralized, allow anonymity, and run autonomously.

In any such decentralized reputational system there is always the potential to corrupt the value of reputation by purchasing it directly, or through automated worthless work, or through the degeneration of a majority of inexpert opinions. These are the respective problems of corruption, Sybil attacks, and tyranny of the majority that have plagued all previous autonomous, decentralized reputation platforms. The architecture solves these problems by creating a crypto economic incentive-based system that promotes domain specific expertise and trust.

The design philosophy is that ideally a person’s reputation is precisely their value. The key, new approach in this platform is the creation of a dynamical evolutionary feedback system which properly incentivizes productive cooperation by tying an expert’s reputation to their proof of ability and productive contributions as verified by validation pools generated by public fees sent to the system.

# 2 Platform architecture

There are four components that constitute the core functioning of the proposed expertise validation platform:

1) A dynamic list of system-generated sub-tokens (called sem tokens), representing reputation/expertise in any domain.
2) The **forum** of expertises. For each expertise, there is a linked list[1] of posts, where each list has a sub-token assigned to it, based on its root post, called the **expertise tag**, or **expertise**. Each post can include opinions, evidence of work, evidence of expertise, policies, and contract templates. A **post** is a trivial smart contract on the Ethereum blockchain, typically a short text post. The forum is housed on a blockchain to allow eternal verification and review of the reputation created in each post.
3) The **bench** of **experts**. Experts are anonymous users who stake their respective sub-tokens to answer validation requests or proclaim their availability for off-platform work.
4) The **validation pool**. Experts may stake their expertise-specific tokens in order to validate or invalidate posts through a betting pool. This is used to answer validation requests, set precedents, promote specialization and proficiency. The validation pool is the mechanism which creates and distributes reputation.

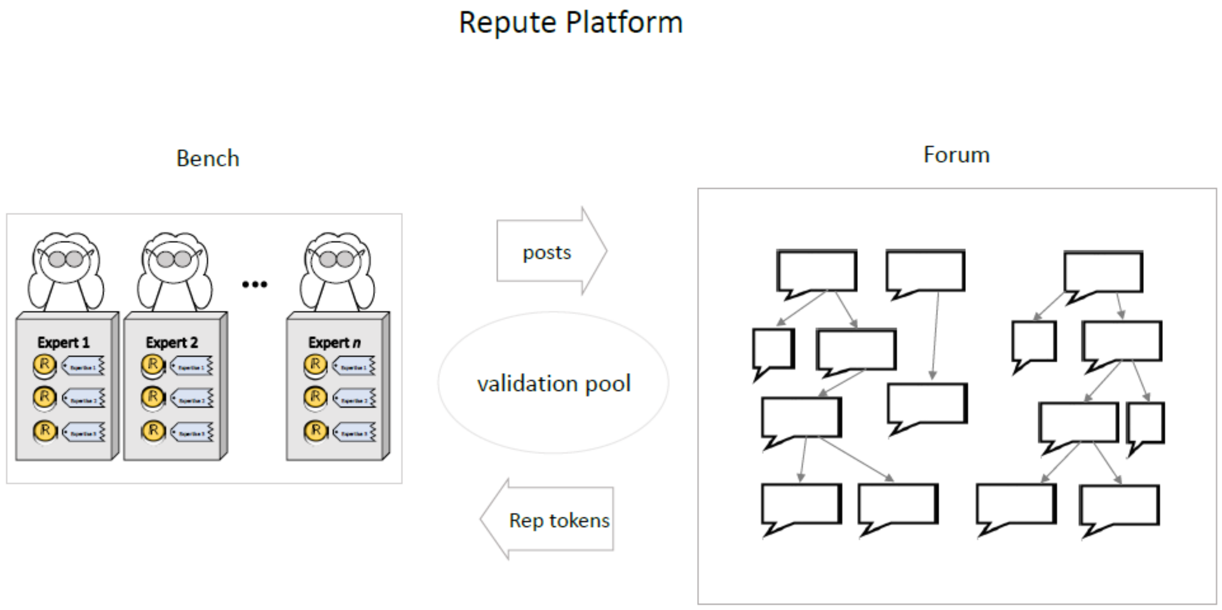


*Figure 1: The platform consists of a collection of experts, called the bench, and an openly readable forum which collects all evidence-of-work posts. A validation pool mediates between the two, determining power, reputation, and precedence in the system.*

Non-expert users, called **public** users, will engage the platform by creating smart contracts which engage the experts from the bench for work off platform. The public is expected to find successful smart contracting language/templates in the forum which the bench experts have created to attract such platform business.

[1] technically a rooted directed acyclic graph, or a citation graph. See §9.2.

N.b.: It is important to realize that this paper focuses on the core architecture design. A typical public user will almost never concern themselves with any of the details explained in this paper. Instead user interfaces (UIs) will be built on top of the core which will guide their experience, facilitating the ability to browse, filter, and analyze the most useful available experts and smart contracts.

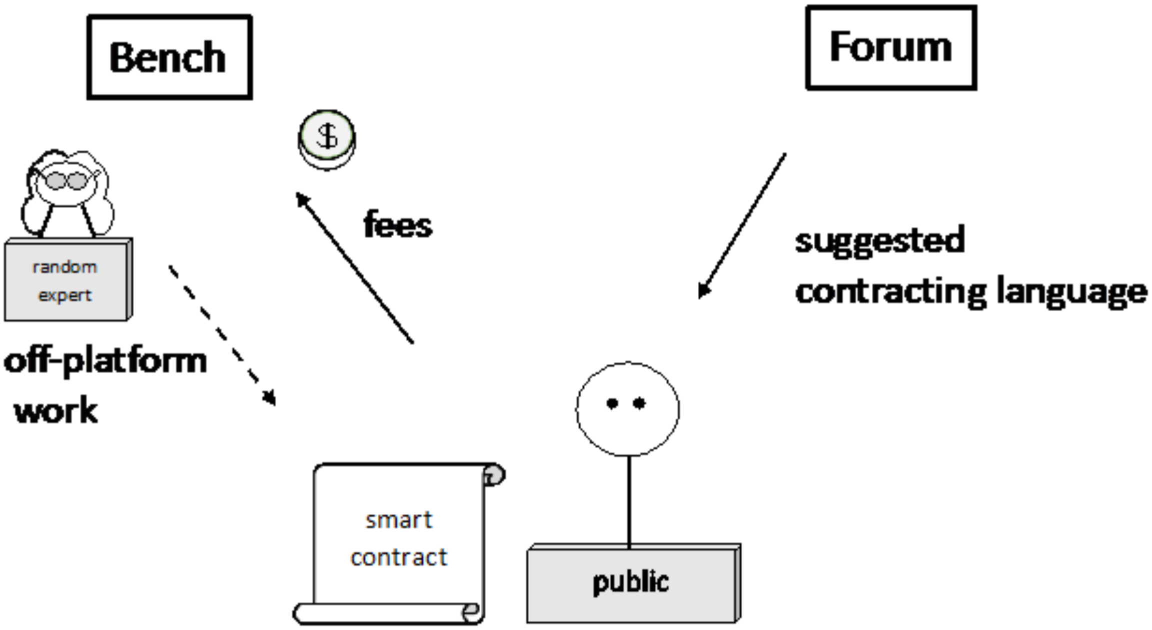


*Figure 2: Public users access the system by calling experts from the bench to perform off-platform work by sending fees to the platform. Public users will find successful smart contracting language for achieving all such aims in the forum, as supplied by the bench experts who develop the forum in order to earn reputation.*

ETH fees from the public are paid to the platform, not the individual experts called. All fees paid to the system accompany a validation request. A **validation request** contains a post and the fee. The **post** is created as evidence of work by the individual expert engaged by the smart contract. The fee is supplied by the public user who engaged the expert for off-platform work. These fees are split as **salaries** amongst all the experts, weighted according to their holdings of expertise tokens. The fees do not go directly to the expert who performs the off-platform work; they are rewarded later as explained next.

If the post is linked to a previous post in the forum, the platform **mints new reputation tokens** associated with the relevant expertise tag[2], in proportion to the fee depending on a pre-set conversion ratio. The reputation tokens are called **sem**. Half the new sem tokens are staked in the poster's name as a bet that the work done is accurate and improves the expertise (this stake is their only direct reward for the off-platform work). The other half of the newly minted sem tokens are staked against the post, and left unassigned. (Figure 3.)

[2] Separate types of ERC-20 tokens are minted for each expertise tag. We suggest the currency symbol SEM -EName for the sem token associated with an expertise named EName.

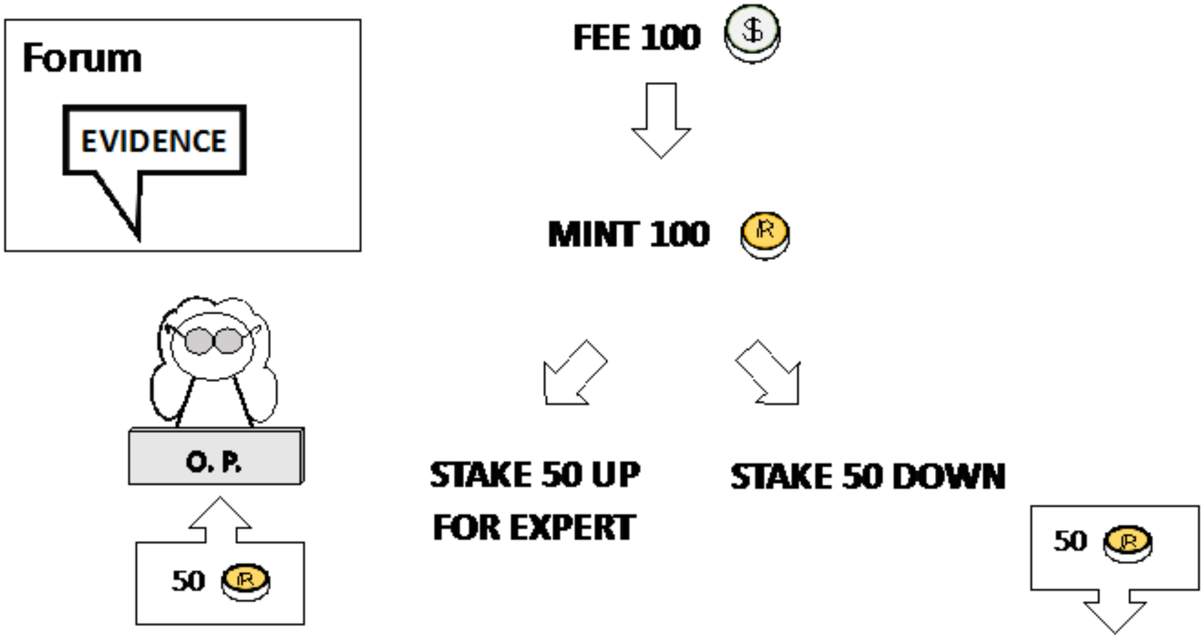


*Figure 3: All fees paid to the system are split amongst all experts, weighted according to their token holdings, as a reputational salary. Individual experts who do the work that attracts the fees are paid indirectly with sem tokens if their evidence-of-work post is accepted by a weighted majority of experts. The original poster (O.P.) sends their post to the forum associated with the fee. The platform mints new sem tokens proportional to the fee. ½ of the newly minted tokens are staked for the O.P. as an upvote, ½ are staked as a downvote, then the rest of the experts are allowed to evaluate the post and participate in the validation pool.*

Every validation request creates a **validation pool**, where experts stake expertise-specific sem tokens to approve or disapprove the post. The winners split the losers' stakes. (Figure 4.) Upvotes win ties. In the event the post loses the validation pool, the half staked against the post which was unassigned is destroyed.

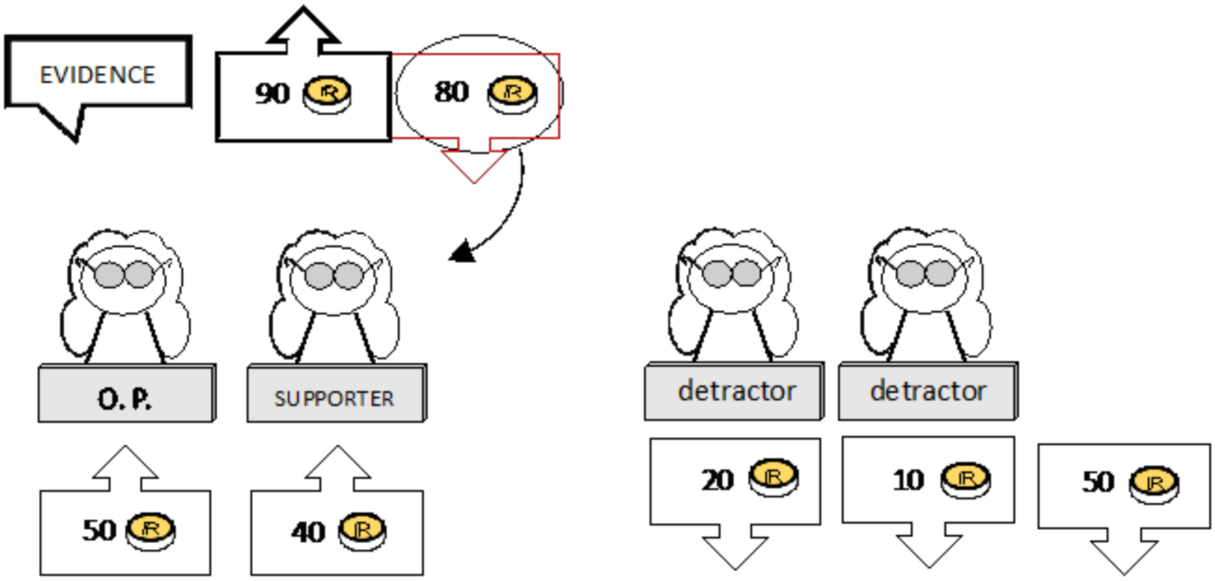


*Figure 4: In this example the upvotes on the evidence-of-work post win with 90 tokens against 80 staked as downvotes. The original poster's (O.P.) newly minted 50 added to a supporter's 40, against the 50 newly minted downvote tokens added to 30 tokens cast as downvotes by detractors. The losers' 80 tokens are split between the O.P., who receives 50/90 of the 80 lost tokens, and his supporter, who receives 40/90. Then the post holds a record of how popular it was amongst experts to determine the post's power as precedence.*

# 3 Practical function

## 3.1 Core basics

While the core architecture can be implemented on other types of infrastructure, Ethereum was chosen to host the first implementation, due to its decentralized, censorship-resistant, modular architecture and extensive developer adoption features.

The core platform exposes five external functions:

1) Get list of expertises.
2) Get list of experts given an expertise.
3) Announce expert availability stake for random selection.
4) Add a validation request to the platform.
5) Vote on a validation request by staking expertise specific tokens.

The typical flow of events for a user interacting with the core is the following:

1) Public user Alice has some work which needs to be done that requires a certain expertise.
2) Alice searches the forum for solutions. An off-platform user interface which organizes the information of the forum suggests the appropriate expertises and smart contracting language for finding an expert in the system.
3) Alice chooses a smart contract (SC), sends SC and the ETH fee to the platform.
4) Accessing the platform, SC does the following:

    a) SC queries the platform to get the list of all experts and their posted availability stakes from the chosen expertise.
    b) SC randomly chooses an expert, Bob. The random selection follows a weighted distribution determined by the posted availability stakes of the experts in the expertise.
    c) SC takes control of Bob's availability stake.
    d) SC informs Bob of his selection and the work Alice requires to be done.
    e) Bob does the work for Alice off platform.
    f) Bob sends his evidence-of-work post to SC.
    g) In Bob's (pseudonymous) name, SC sends Bob' evidence-of-work post and Alice's ETH fee to the platform.

5) The platform distributes Alice's ETH fee to the domain experts, proportionately, based on the amount of sub-tokens they own for that expertise
6) The platform mints new domain-specific sub-tokens, based on the amount of ETH sent and the exchange rate between ETH and the respective sub-token. (Within each expertise, the bench decides the exchange rate. See §9.1 Expert parameter maintenance.)
7) Validation pool opens. Half of the new tokens are staked for upvote and assigned to Bob. (This is Bob's payment for the work he's done). The other half of the new tokens are staked for downvote and left unassigned. (Figure 3, above.)
8) SC stakes Bob's availability stakes as an upvote for Bob. (This incentivizes Bob to work well for Alice, or he will lose his availability stakes.)
9) Validation pool is announced to all experts to vote on. The staking process will remain open for a limited amount of time, set as a parameter. (Within each expertise, the bench decides this expiration time. See §9.1 Expert parameter maintenance.)
10) During this time, any other domain-specific sub-token holders will have the opportunity to stake their tokens as up- or downvotes, using symmetric key encryption[3] to hide their vote from the other experts. (Figure 4, above.)
11) When the validation pool ends, voting parties send their decryption keys to reveal up- and downvotes. The winning parties will be awarded their staked amounts plus the pro-rata share of the losing party's stakes. If Bob's work is validated, he will be paid in tokens in proportion to the ETH staked in his name, and his pro-rata share of the losing party's token stakes.

In summary, the core

1) answers queries about the list of expertises, the list of experts within each expertise, and each expert's availability stakes
2) answers validation requests, taking as input an ETH fee and the address of an Ethereum post, then
    a) distributes the ETH as reputational salaries to the bench
    b) opens a new validation pool
    c) collects reputation stakes from experts on the pool
    d) distributes the losers' stakes to the winners

[3] This symmetric encryption protocol must be chosen carefully to avoid the birthday problem attack, where malicious voters could send an encrypted key that could be decrypted in two different ways.

## 3.2 Details

### 3.2.1 New experts

To become a **new expert**, a public user Alice posts a comment in the forum with a pointer to a previous post in the expertise and adds a fee her name. Following the protocol for whenever a fee is added to the platform, new tokens in the expertise tag are minted. Half the new tokens are staked for Alice; half are staked against. Experts evaluate her post by staking their reputation in the betting pool. If Alice's comment wins, she is vested by the validation pool with reputation in the expertise linked to Alice's pseudonymous public key identifier, and she becomes a new expert. If Alice's comment loses, she loses all her reputation in the expertise and returns to the public.

The 50/50 stake guarantees a new expert cannot simply buy expertise, as the previous experts have complete power to decide whether the new expert has contributed positively to the platform.

### 3.2.2 New expertises

To create a **new expertise tag**, a public user Adam posts a new root comment in the forum and adds a fee in his name. The system then follows protocol, automatically vesting Adam with reputation in the new expertise tag. Specifically, new sem tokens are minted in the amount of the fee chosen. Half the new tokens are staked for Adam; half are staked against. Since no other experts exist to evaluate the post, the Adam's comment wins the tie, and he is vested by the betting pool with reputation in the expertise.

Adam then has complete power to add further posts linked to this to the forum, improving the expertise tag with mission statements, protocols, evidence of expertise, and advertisement, to encourage new experts to join and eventually attract public fees for off platform expert work. Adam has complete power to accept or reject the second expert to join the expertise's bench.

The forum will serve as a testing ground and evolutionary ecosystem for proving good practices within each expertise. In particular, to encourage public use of the expertise, smart contracting template code should be suggested by experts and continually subjected to the improving process of critical comments. Policies, evidence of experience, evidence of successful work, and advertisement for services will also be continually added and improved in the forum.

### 3.2.3 Platform economics

The **economics** of the platform is simple. All of the currency added to the platform comes from fees from public users submitted associated with a post, the inputs of a validation request. All currency taken from the platform goes to experts through reputation-weighted salaries.

### 3.2.4 Contract freedom and fees

Validation fees and the method of expert selection are decided by the smart contracting parties. Whether the work requested and the protocols included in a smart contract are accepted is decided by the experts selected. The means for successful cooperation between the public and experts will evolve with continually improving smart-contracting language in the forum, such as appropriate fees and standards of work.

To allow maximum freedom for users, the core does not require specific smart contracting language from the public users to engage a validation pool, simply a fee and the address of a post. So not all of the events in the "typical flow of events for a user interacting with the core" listed above are necessary. In some cases (e.g., §7.3 Example: Company organization) these events will not always be used.

However, the platform was designed to thrive in an environment with anonymous, potentially hostile users, *if* the following choices are chosen to be included in all smart contracts which use experts for off-platform work:

A. Experts should signal their availability by staking sem tokens (called an **availability stake**).

B. Experts are selected randomly, weighted by reputation according to their availability stakes (see Figure 5).

C. The availability stake should automatically be added to the expert's evidence-of-work post as the expert's upvote stake in the validation pool.

D. All fees should be sent to the platform for the work done.

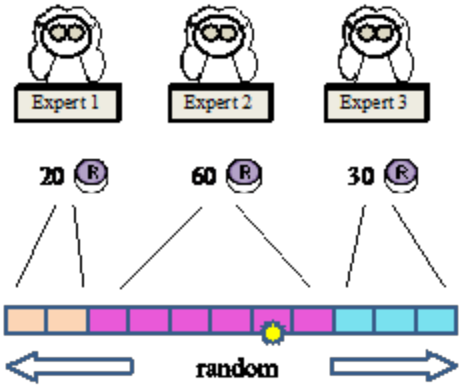


*Figure 5: Random selection of experts is decided by relative weight of reputation. Before a smart contract is engaged, experts have the opportunity to stake sem tokens to signal their availability for work. These availability stakes will be added as the chosen experts' upvote bet on their evidence-of-work post. In this example the yellow star stops randomly along the bar, but is most likely to stop on the 2nd expert.*

For the healthy development of the platform, public smart contracting parties are expected (but not required by the platform) to choose to add such stipulations to their smart contracts to engage work from the platform. They are expected to do so because evidence is posted to the forum that such measures achieve the relevant goals. Experts are expected to police this behavior by refusing to engage in smart contracts without the proper protocols. In that case the ETH fee included in the smart contract would automatically be distributed to the experts who policed the action by downvoting the post, punishing the public user who sent the fee and the faulty contract for wasting the bench's time.

One counterintuitive element is that the chosen expert is not paid directly in fees, only indirectly in newly minted tokens. The platform was designed to ensure security, and encourage maximal productive cooperation, by valuing reputation over all else. A crucial element of ensuring this security and cooperation is therefore to require users who want validation to send *all* work fees to the platform to earn maximal reputation. Off platform work that is not submitted for evaluation from the experts is to be discouraged. This protocol should be established in the forum, and the vested experts should police this behavior by automatically downvoting any validation request which doesn't send the whole fee, so that people will never attempt such behavior in their expertise tag.

Choosing to enforce a protocol where all fees are shared with the expertise creates a successful positive feedback loop: The more fees are sent to the platform, the more the reputation is worth; which means the sem tokens will be more desirable to experts than one-time fees; which means the system will be carefully policed by vested experts; this makes the system more secure; so the platform will attract more public fees. Cf. Figure 6, below. This makes a tag which enforces the protocol more competitive than a tag which doesn't.

Further, whatever contributions experts provide to the system will be rewarded appropriately eventually, because all of the fees that the expertise attracts are paid to the experts according to their token holdings. None of the fees are taken by any centralized authority; there is no outside owner of the platform. "Objective expertise" is ultimately measured by the public fees the expertise tag attracts.

In review, experts from the bench stake their expertise specific sem tokens to 1) evaluate posts with upvotes or downvotes, and 2) announce their availability for working public smart contracts by posting stakes of their reputation. The forum evolves to include information on 1) proper smart contracting language for the public to use to engage experts, including appropriate fees and protocols for work, 2) validation methods, policies and principles for the future development of the expertise tag, and 3) reputation-verified evidence of expertise through posts of successful work. The validation pool intermediates between the bench and the forum to vest experts with verified reputation, which is valuable as power to influence the development of the expertise and valuable in gaining future reputational salaries.

With these solutions the platform automatically scales to be able to handle any type and value of work, as illustrated in §7 Examples.

# 4 Problems solved

In this section we present the problems solved by the platform and substantiate the claims made in the abstract.

## 4.1 Attack resistance

Traditionally, the most dangerous attacks on reputation-based decentralized autonomous platforms have been Sybil attacks, tyranny of the majority, and the 51% attack. In this section we discuss how the platform inhibits these attacks. How the platform inhibits further types of attacks is discussed in the appendix, §8 Critical analysis.

### 4.1.1 Sybil attacks

Simple Sybil attacks consist of any anonymous single user opening any number of expert accounts to use worthless automated work in order to unjustly profit from the system. The platform is Sybil attack resistant because all power in the platform is weighted by reputation. An owner of a single account with 1000 tokens has the same or more power than an owner of 1000 accounts with one token.

The power of reputation in the platform has 3 uses: being chosen as an expert for off platform work, voting on posts, and salaries. All of these actions are unaffected or weakened by spreading reputation between accounts. Thus, simple Sybil attacks are prevented.

Since fees are paid indirectly as salaries, the platform encourages development of reputation. Since all newly-minted sem tokens join the platform as 50/50 up- and downvote bets, there is little opportunity to overwhelm the system with outside power, since established experts always have complete control to decide whether new reputation is given to a user. Thus reputation is only ever vested to those who can prove to existing experts that their contributions improve the platform.

### 4.1.2 Tyranny of the majority and tragedy of the commons problems

Another significant threat to fair and just decision-making in an anonymous and democratic reputational system is the potential for tyranny of the majority. Alexis de Tocqueville coined the term tyranny of the majority in order to describe the corruption in a system that manifests itself in decisions made with a greater concern for following what is perceived as popular than what is right and just.

Similarly, the tragedy of the commons occurs in any system which does not have well designed incentive structure. This is called the “nothing at stake” problem in blockchain proof of stake design, where unregulated systems lead pseudonymous users to abuse the system.

As an example of tyranny of the majority and tragedy of the commons problems, spend an afternoon reading YouTube video comments. As a second example, Reddit’s upvote system currently has insignificant punishment for voting randomly, which compromises the value of the meaning of an upvote[4].

There are three ways we prevent tyranny of the majority in the platform. First, in a weighted voting system greater power generally accrues to those with greater expertise. Then, in the worst-case scenario, voters will be concerned with voting in line with powerful experts, which generally make valuable decisions and favor precedent and predictability.

The second mechanism that counters the tyranny of the majority is the time-delay in announcing the results of upvotes, so experts cannot plainly see the majority position before posting their stake.

[4] Maria Glenski, Corey Pennycuff, & Tim Weninger, “Browsing and Voting Patterns on Reddit”, IEEE Transactions on Computational Social Systems, Volume: 4, Issue: 4, Dec. 2017, pp. 196 - 206, 6 September 2017, Available: https://arxiv.org/pdf/1703.05267.pdf

Finally, any expertise tag which is corrupted will not attract fees, and it is easy to create a competing expertise tag to replace it.

### 4.1.3 The 51% attack

One way to game the system is for a group of malicious actors who hold 51% of the tokens to collude and vote against common sense, thus taking a good percentage of the community's resources. In an open system, however, such an attack would be quickly apparent. This would erode trust in the platform, and therefore use and fees would diminish, making the 51% holding of reputational salary less valuable. Therefore this tactic generally has the potential to destroy the expertise tag, but not enrich the attackers.

However, it is feasible that an arbitrage opportunity could evolve if the experts do not police their expertise. If a significant percentage of the (technically fungible) tokens were put on sale in a token exchange, a malicious actor would have the opportunity to 1) buy 51% of tokens, 2) override the bench by voting against common sense in a popular betting pool, taking a significant amount of tokens, 3) sell the tokens quickly, making a profit before the tokens lose their value due to the attack.

This attack can certainly occur in theory. However the system naturally guards against such an attack. First, it is difficult to sell off 51% of anything in a short period of time. In an open system, it is unlikely the attacker could sell all 51% of the tokens immediately after ruining the trust of the system by voting against common sense. It is extremely likely in this scenario that the 51% will lose more value than would be gained.

Secondly, it is unlikely that a malicious group will ever gain 51% of tokens. Every token ever created is initially given to someone who has demonstrated their expertise and investment in the platform. While such tokens are fungible and there are circumstances where they will be sold, healthy expertise tags will not leave significant tokens in disuse on the market. (A healthy expertise tag is defined as one which is generating fees.) As demonstrated below in §4.4.2.2 the value of sem tokens are primarily in their use—tokens lose relative value as fees are added to the system. Tokens are more valuable to those global users who truly have the expertise for doing the relevant work and policing, than they are for people who wish to own them as an investment without using them. In a healthy expertise tag, tokens will continually be used to evaluate posts to earn more sem tokens and the reputational salary.

Therefore the total of all tokens not in use by actual experts, will not likely ever amount to 51% in a health expertise, since new tokens are minted continually and used by those who earn them more than they will be sold.

A malicious actor could attempt to gain 51% of tokens over a long period of time, as they become available. But there are two problems with this strategy. First, because new tokens are being minted and used all the time, they will likely never amount to 51% in a healthy expertise, even if tokens are continually added to exchanges. Secondly, if the expertise is not healthy enough to generate more tokens which are used than are sold on exchanges, buying tokens on an exchange over a long period of time will be very expensive because of the aforementioned natural inflation. So the arbitrage opportunity doesn't exist unless a single transaction is worth much more than the entire reputational value of the expertise. And if a malicious actor wishes to destroy a tag, they must spend much more than the entire reputational value of the expertise. Healthy expertises are secure because it is easier to profit from them by improving them than by harming them.

The other way to gain 51% of the tokens is to pay fees to the system and earn new tokens by winning betting pools. As demonstrated next in §4.1.4, a reasonable estimate for the price of this attack (even when the bench makes absolutely no effort to police this action) is 6 times the value of the entire quantity of sem tokens in existence in the expertise, when the expertise is healthy, i.e., earning fees. At the same time this action would reward all the good faith actors in the system beyond what they invested.

In either case, if an attacker does not acquire 51%, it is extremely risky to attempt the attack, as all their tokens are likely to be lost when betting against common sense. And if an attacker does not acquire 51%, they will lose a significant amount of their value to the natural inflation of the coin while they are attempting the attack.

**Most importantly, a key benefit of the system is that as it is used, it becomes measurably more secure.**

### 4.1.4 Reputation calculation

Here we demonstrate the attack resistance of the platform by showing that in the absolute worst-case scenario, the price to corrupt the system from within is a minimum of twice the total historical fees added to the system. If any obvious protections are instituted, the price to corrupt grows steeply.

Consider the situation where the platform has a total of $g_0$ sem tokens at time 0. We will refer to all these users as good-faith experts[5]. Imagine a malicious group $m$ wishes to purchase 51% of the reputation with fees, either to profit from future transactions or to destroy trust in the particular expertise tag. Since only half of their fees are added as sem tokens for the malicious group, and the other half will be added to the existing good-faith actors it is difficult to achieve the malicious goal, even in this worst-case scenario.

Under the **worst-case scenario**, we suppose there are no safeguards against joining as an expert—any fee of any size is automatically accepted in the validation pool, and resolved instantly with no time delays. Finally we assume no other users are adding any fees.

Under this scenario, the best strategy for the malicious group to gain reputation by paying fees is to make many micro-payments. In the same way that continuously-compounded interest is better than discrete interest at the same rate, micro-payments allow malicious users to profit off their own later fees once their earlier fees vest them with reputation.

Thus we assume the malicious group $m$ will add a variable number $n$ of fixed fees of size $\Delta x \ll 1$. Then the good faith experts will have

[5] The good-faith experts do not need to behave altruistically; they are simply required to not collude with the new, malicious group and to act in their own self-interest.

$$g_{n+1} = g_n + \frac{1}{2}\Delta x(\frac{g_n}{g_0 + n\Delta x})$$

reputation after $n + 1$ fee payments from the malicious users, with $g_0$ being the initial reputation of the platform, while the malicious user will have

$$m_{n+1} = m_n + \frac{1}{2}\Delta x(\frac{m_n}{g_0 + n\Delta x}) + \frac{1}{2}\Delta x$$

sem tokens, with $m_0 = 0$. Therefore

$$\frac{\Delta g_{n+1}}{\Delta x} = \frac{1}{2}(\frac{g_n}{g_0 + n\Delta x})$$

which becomes the ODE

$$\frac{dg}{dx} = \frac{1}{2}(\frac{g}{g_0 + x})$$

as $\Delta x \to 0$. Similarly

$$\frac{dm}{dx} = \frac{1}{2}(\frac{m}{g_0 + x}) + \frac{1}{2}$$

We want to know how many fees $x$ are required before the malicious group can overwhelm the system with 51% reputation power, so we solve $g(x) = m(x)$ for $x$.

Solving the ODEs using the method of integrating factors gives the formulas

$$g(x) = \sqrt{{g_0}^2 + g_0 x}$$

and

$$m(x) = x + g_0 - \sqrt{{g_0}^2 + g_0 x}$$

So $m$ overwhelms the system once $x = 3g_0$.

However, while the malicious users are paying their fees, they are also receiving salary payments that are increasing as they gain reputation. While paying the $3g_0$ in fees, the total salary regained by the malicious group is

$$\int_0^{3g_0} \frac{m(x)}{m(x) + g(x)} dx = g_0$$

Consequently the malicious group would need to invest an absolute minimum of $2g_0$, that is, double the total reputation of the system to gain 50% power in the system in order to outvote the rest of the good-faith experts in the validation pool.

We stress that the value of $2g_0$ is an extremely conservative lower bound. In practice, the investments would need to be discrete values, not continuous. This significantly raises the minimum corrupting investment, and significantly lengthens the time needed to gain 51% power. Also, in practice many other users will be paying fees besides the malicious group, especially if the malicious group is pumping fees into the system.

Assuming the malicious group does manage to take over the platform, the rest of the world would immediately see the unfair action that couldn't be rectified by the good-faith actors, so the expertise tag would topple, losing its value. The malicious group would merely gain the (fraction) of the fee from one transaction. As long as any single fee is smaller than the total reputation, there is no incentive to game the system for fees.

The only other reason to act maliciously is to destroy the particular expertise. But then the $2g_0$ invested becomes worthless. So the minimum cost at any time to topple the system is twice the total sem tokens—assuming you can simply buy reputation without limit or oversight.

Therefore as long as the fees are low compared to the reputation total, or if there is no competitor that is suffering by the platform's existence by more than twice the total number of sem tokens, it's not worth spending money to corrupt the system—it is more valuable to use the power to improve the platform.

Further, as fees are added, reputation is added, so it becomes more secure as time goes on, even assuming constant fees.

Further, if a malicious group wishes to strike, their best strategy is to strike immediately. Their corrupt sem tokens become less valuable if they don't use them, as each new fee they didn't add to the system adds to other peoples' tokens, diminishing the value of older tokens. So even if there are many different malicious groups, if no single malicious group gains 51% control, the system cannot be corrupted to unfairly earn fees.

A malicious group is therefore limited to attempting to destroy the platform, by overwhelming it with fees. If that is happening, and there is a reasonable time-delay in resolving betting pools, then the incoming malicious fees would encourage other, non-colluding parties to invest as well. This would significantly slow the effort to gain 51% for the malicious group.

Specifically, assume the good-faith experts invest some constant fraction $c$ of the malicious groups' investment $\Delta x$ at each time. If $c \geq 1$then the malicious group will never gain more than 50% power. So assume $0 < c < 1$. Then the equations become

$$\frac{dg}{dx} = \frac{1}{2}(1+c)(\frac{g}{g_0 + (1+c)x}) + \frac{1}{2}c$$

and

$$\frac{dm}{dx} = \frac{1}{2}(1+c)(\frac{m}{g_0 + (1+c)x}) + \frac{1}{2}$$

which are solved to give

$$g(x) = \frac{\sqrt{g_0 + (1+c)x}}{1+c}(c\sqrt{g_0 + (1+c)x} + {g_0}^{1/2})$$

and

$$m(x) = \frac{\sqrt{g_0 + (1+c)x}}{1+c}(\sqrt{g_0 + (1+c)x} - {g_0}^{1/2})$$

Then the solution to $g(\underline{x}) = m(\underline{x})$

$$\underline{x} = [\frac{4}{(1-c)^2} - 1]\frac{g_0}{1+c}$$

is the total amount the malicious group $m$ must invest, and they recover

$$\int_0^{\underline{x}} \frac{m(x)}{m(x)+g(x)}dx = \frac{1}{1+c}\left(\underline{x} - \frac{2{g_0}^{1/2}}{1+c}[\sqrt{g_0 + (1+c)\underline{x}} - \sqrt{g_0}]\right)$$

of those outlays through reputational salary. So the final cost to override the system when good-faith experts are investing the fraction $c$ of the malicious investment is

$$[\frac{4}{(1-c)^2} - 3]\frac{g_0 c}{(1+c)^2} + \frac{2{g_0}^{1/2}}{(1+c)^2}\sqrt{g_0 + (1+c)\underline{x}}$$

For instance, if the malicious group is doubling the investment of the rest of the community, then $c = 1/2$ and $\underline{x} = 10g_0$of total reputation must be spent in fees, while more than $\frac{58}{9}g_0 \approx 6.44g_0$ will be lost.

To improve simulations, we can also solve the recurrence relation directly, without moving to the ODEs, since the recurrence relation is first order and linear:

$$g_{n+1} = g_n + \frac{1}{2}(1+c)\Delta x(\frac{g_n}{g_0 + n(1+c)\Delta x}) + \frac{1}{2}c$$

has solution

$$g_{n+1} = \frac{g_0 c}{(1+c)\Delta x} + nc + \frac{((1+c)\Delta x - c)\Gamma(1 + \frac{g_0}{(1+c)\Delta x})\Gamma(n + \frac{1}{2} + \frac{g_0}{(1+c)\Delta x})}{\Gamma(\frac{1}{2} + \frac{g_0}{(1+c)\Delta x})\Gamma(n + \frac{g_0}{(1+c)\Delta x})}$$

where $\Gamma$ is the gamma function, while

$$m_{n+1} = m_n + \frac{1}{2}(1+c)\Delta x(\frac{m_n}{g_0 + n(1+c)\Delta x}) + \frac{1}{2}\Delta x$$

has solution

$$m_{n+1} = \frac{g_0}{1+c} + n\Delta x - \frac{\Delta x \Gamma(1 + \frac{g_0}{(1+c)\Delta x})\Gamma(n + \frac{1}{2} + \frac{g_0}{(1+c)\Delta x})}{\Gamma(\frac{1}{2} + \frac{g_0}{(1+c)\Delta x})\Gamma(n + \frac{g_0}{(1+c)\Delta x})}$$

We further discuss the consequences of these calculations in the appendix §7.4.1: Proof of Stake.

## 4.2 Countering corruption

The platform offers an infrastructure of crypto-economic incentives that encourages productive contributions to the platform and inhibits malicious conduct.

The platform's structure creates an incentive-driven positive feedback loop. (Figure 6.)

1) Between the forum and the public, the forum must provide useful smart contract templates, evidence of platform expertise, and evidence of past successful business in the forum. Then the public will engage the bench with fees and smart contracts.

2) Between the public and the bench, the public must choose the proper smart contracting language (including proper fees and type of work), when engaging an expert. Then the bench will improve the forum with evidence of successful work.

3) Between the bench and the forum, experts must improve and police the forum through evidence of successful work verified in a fair validation pool, and by adding useful smart contracting templates and effective policies. Then the forum will give the public the confidence to write smart contracts.

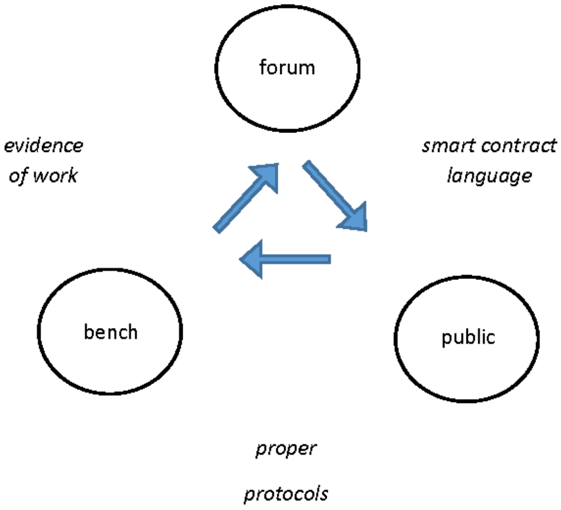


*Figure 6: Feedback loop incentivizing productive collaboration.*

The process of interaction between the branches is designed to be fairly balanced. Since all newly minted tokens are staked 50/50 for and against the post and all losers' tokens are split by the winners, the system encourages careful and judicious self-policing. This 50/50 balance encourages unbiased and truth-seeking voting, since neither side is favored. Since all fees are split between sem token holders, the entire incentive in the system is to acquire reputation, balancing the motivations of protecting the value of reputation in the expertise tag and earning new fees for the expertise tag. Finally since all fees that come into the platform are disbursed to the expert users and not any asymmetrically powerful outside owners of the platform, the system is internally balanced against outside corruption.

Therefore the feedback loop of the platform is **closed**, in the sense that there are no rent-seeking[6] owners: The system is entirely supported by users, and the users reap the entire profit.

The anonymity of the experts and the openness of the public posts of work evidence in the forum focuses the energies of users on the true essence of reputation—i.e., ability and positive contributions—while discouraging the corruption associated with identity and rent-seeking behavior.

## 4.3 System autonomy

The platform achieves full autonomy almost immediately. Once the Ethereum DApp is posted to the blockchain, its authors have no more control over the evolution of the expertise tags in its forum than any other Ethereum user. Creation, hosting, and maintenance costs are borne entirely by users through fees paid to the platform, which are distributed to the expert users after paying Ethereum gas fees.

See §9.1 for a discussion of how experts adjust core parameters within each expertise.

## 4.4 Reputational meritocracy

### 4.4.1 Fair distribution of power

The platform encourages efficient collaboration through accurate distribution of power according to reputation. The only rewards possible are received for expert-evaluated improvements to the platform. The system is designed to resist economic corruption and inefficiencies by fairly evaluating all work through the measured vesting of valuable reputation.

The forum is the environment where experts demonstrate their value. Experts gain reputation by posting and evaluating posts. Successful posts are comments in the forum which win the upvote validation pool. Winning the validation pool means the other experts evaluate the post and believe it improves the value of the expertise. Types of posts include:

- policies for future development of the expertise
- protocols for acceptable behavior

[6] From economics, the term "rent-seeking" means the use of resources to enrich a party without creating any wealth for society in return.

- evidence of experience
- evidence of work
- creating or improving smart contracting templates
- criticizing any of the above

Upvotes naturally determine the strength of precedence set by each comment. (See §9.2 The forum as a weighted DAG for a more nuanced explanation of the power of precedence.)

## 4.4.2 Platform economics

### 4.4.2.1 Basic salary definition

The platform is decentralized and autonomous. There is no centralized foundation which owns any part of the platform. So the economics is very simple.

All money into the system comes from fees paid by business access, and fees from commenters who post to become new experts. New commenter fees are expected to be smaller than outside business fees in the long term for healthy expertise tags; otherwise the expertise tag is simply a pyramid scheme.

All money out of the system is paid as salaries to experts according to reputation. All fees paid to an expertise tag are split between the experts weighted according to their relative holding of sem tokens in that expertise tag.

Specifically, supposing expert $E_i$ has reputation in only one expertise tag, then their salary $S(E_i)$ is given by the formula:

$$S(E_i) := \frac{R(E_i)}{\sum_j R(E_j)} \sum_k F(x_k)$$

where $R(E_i)$ is the total sem tokens of expert $E_i$ so $\sum_j R(E_j)$ is the sum of the reputations of all experts $E_j$ in the expertise tag, i.e., the total number of sem tokens in that tag; and $F(x_k)$ is the fee paid that is associated with post $x_i$ so $\sum_k F(x_k)$ is the total fees paid into the system allocated to that expertise tag during that pay period.

Experts with sem tokens in several expertise tags are paid as specified above for each tag.

A slightly more complicated formula is available for determining salaries when tokens are weighted according to a reference scheme as in §9.2 The forum as a weighted DAG.

### 4.4.2.2 More reputation calculations

In this section we illustrate the value of one sem token with a few calculations. This demonstrates the value of an investment in reputation in the platform without using its power to evaluate posts. The conclusion is that the economy is inflationary at equilibrium, which improves its security and discourages rent-seeking.

**One sem token loses value if fees are constantly added in time.**

If fees are constantly being paid into an expertise tag at a rate of $F(t) = r$ units per time, then the salary paid for one coin decreases in time. To see this, denote the total number of sem tokens in the tag at time $t$ by $T(t)$. The salary per token per unit time is $S = F/T$. However, since each fee token creates a new sem token, we have

$$T(t) = T(0) + rt$$

so

$$S(t) = \frac{r}{T(0) + rt}$$

which shrinks to 0 as time goes on.

**One sem token loses value if fees grow linearly in time.**

If the fees are given by $F(t) \coloneqq ct$ then the reputation recurrence relation $R(t+1) = R(t) + F(t)$ can be solved to give

$$R(t) = R(0) + F(0) + ct(t-1)/2$$

so

$$S(t) = \frac{ct}{R(0)+F(0)+ct(t-1)/2}$$

which shrinks to 0 as time goes on.

**One sem token pays constantly if fees grow exponentially in time.**

If the fees are given by $F(t) \coloneqq F(0)e^{ct}$ then reputation satisfies the recurrence relation $R(t+1) = R(t) + F(t)$ which may be solved to give

$$R(t) = R(0) + F(0)\frac{e^{ct}-1}{e^{r}-1}$$

so

$$S(t) = \frac{F}{T} = \frac{F(0)e^{ct}}{R(0)+F(0)\frac{e^{ct}-1}{e^{r}-1}} \to e^{r} - 1$$

as $t \to \infty$.

**Early sem tokens are more valuable than later tokens.**

All sem tokens in a single expertise tag are equal at any given moment[7]. However, assuming a steady state rate of fees, the payout for 1 year from inception is greater for tokens minted earlier,

[7] Relative to their citation weight. See §9.2 The forum as a weighted DAG.

because later tokens are a smaller percentage of the total. Remember, new tokens are created with every fee paid into the platform, but never destroyed.

This may mean later experts have less motivation to join if fees paid into the system are at a steady state. To combat this, the bench may choose to change the exchange rate between fees and sem tokens to encourage new recruits.

# 5 Off-platform user interfaces

Separate, off-platform entities will emerge to provide systems for connecting users with the platform. They will provide solutions for improving user experience in interacting with the core platform. These off-platform entities will organize the information created in the platform, such as editing the post forum to be relevant to their particular users, suggesting smart contracting solutions for business, suggesting work for experts outside and inside the platform, and streamlining education in obtaining experience and reputation with the platform.

# 6 Summary

The platform is a decentralized autonomous platform for validating domain specific reputation (expertise). The platform includes 4 main components:

1) A dynamic list of system-generated sub-tokens, representing reputation/expertise in any domain.
2) The **forum** of expertises. For each expertise, there is a linked list of posts, where each list has a sub-token assigned to it, based on its root post, called the **expertise tag**. Each post can include opinions, evidence of work, evidence of expertise, policies, and contract templates. A **post** is a trivial smart contract on the Ethereum blockchain, typically a short text post.
3) The **bench** of anonymous experts/validators who stake their respective sub-tokens to answer validation requests or proclaim their availability for off-platform work.
4) The **validation pool**. Experts may stake their expertise-specific tokens in order to validate or invalidate posts through a betting pool. This is used to answer validation requests, set precedents, promote specialization and proficiency.

Public users access the core by requesting a validation pool, which requires an arbitrary post and fee as input. The public is therefore free to choose their own smart contracting language for employing the platform. Public users control all contractual choices, including deciding which types of expertise are required from the bench, what fees are paid to the system, and how experts are called. Experts decide whether such fees and work are acceptable. The forum mediates between the two.

The commenting forum will also be a testing ground for improving smart-contracting language via the upvote process of validation pools. Successful solutions will evolve via the criticism of actual and hypothetical contracts.

A validation pool is a betting pool, where experts stake percentages of their total reputation to up- or downvote posts in the forum. The winners divide the losers' stakes, enriching their reputation.

Reputational tokens are minted in the core with each fee paid by the public, proportional to the fees paid by the smart contract. These tokens are automatically staked fairly as 50/50 up- and downvote bets on the associated posts to ensure fees never influence decisions in validation pools.

Fees paid by smart contracts employing the platform are distributed to all experts in salaries weighted by reputation.

Combined with the platform's two concrete branches, the bench of experts and the forum, the third branch of the system, consisting of the public smart-contracting parties, creates an incentive-driven feedback loop entailing a system of checks and balances which combat corruption and encourage healthy development of the platform and the larger crypto economy.

The platform encourages a reputation-driven meritocracy, since global users with genuine expertise will find it easier to gain and maintain sem tokens than those who buy them. As demonstrated in §4.4.2.2 sem tokens lose relative value as fees are added to the system, leading to inflation, which discourages rent-sitting.

This meritocratic system therefore inhibits the 51% attack. The 50/50 staking of all fees which enter the system, and vesting of reputation according to a fair validating pool from reputation-weighted users inhibits Sybil attacks and the tyranny of the majority problem.

The system automatically scales to valuate any type of expertise with verified and regulated reputation.

*Special thanks for technical input: William Long, Axel Boldt, Greg Waring, Zach Smolinski, Anita Milanovich*

*Technical terminology defined: bench, forum, public, fee, post, expertise tag, expertise, sem token, fee, upvote, betting pool, availability stake, salary*

# 7 Appendix: Examples

To aid imagination, this section details some of the expertises that will be developed in the forum.

First we discuss how a general expertise in the gig economy is developed by arbitrarily choosing carpentry to illustrate how experts can organize and cooperate to attract public fees for their work. Second, we give a more detailed illustration of how dispute resolution can be achieved for general smart contracts in an expertise we call Distributed Jurisdiction. This expertise will be invaluable to the evolution of the platform, as it will give users insurance their smart contracts will perform as expected. Third, we show how an entire company can be organized on the platform. Finally we give one example of how is capable of hosting automated work, with a proof of stake application, oracles, and machine arbiters.

## 7.1 Example: The gig economy

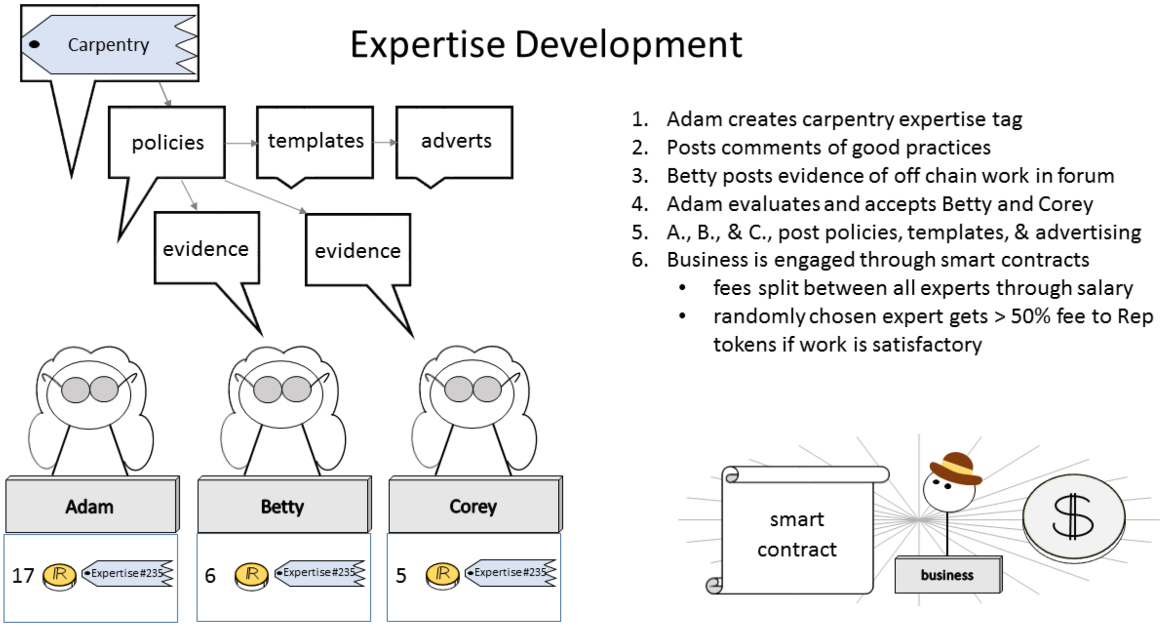
Consider the example of a carpenter, Adam, who would like to add a carpentry expertise tag to the system.

Adam uses ETH to create the expertise tag. Being vested with sem tokens, Adam continues to make comments in the carpentry expertise tag, elucidating policies and advertising to attract further experts to join the expertise.

Next Betty applies to gain carpentry sem tokens by posting evidence of her skills and work with an ETH stake. Adam evaluates Betty's posts and decides whether to allow her to join. If successful, Betty gains sem tokens which she can use to contribute to the carpentry expertise tag by:

- evaluating future applicants
- improving the carpentry forum by adding and improving
    - policies
    - evidence of expertise
    - smart contracting templates for engaging the services of the bench of experts
    - advertising to outside parties
- and posting her availability for work

Finally public users can engage the platform using the smart contracting templates provided by the bench experts in the forum. Public users can set bounties for work they would like to have done off chain, and experts can pick up the work according to their availability. The evidence of past successful performance in the open forum, as verified by the results of blockchain-certified validation pools, will give the public the necessary assurance their smart contracts will perform as expected.



1. Adam creates carpentry expertise tag
2. Posts comments of good practices
3. Betty posts evidence of off chain work in forum
4. Adam evaluates and accepts Betty and Corey
5. A., B., & C., post policies, templates, & advertising
6. Business is engaged through smart contracts
   - fees split between all experts through salary
   - randomly chosen expert gets > 50% fee to Rep tokens if work is satisfactory

*Figure 7: The platform facilitates the development of any expertise in the gig economy through the establishment of verified reputation. Smart contracting language and past evidence of their success found in the open forum will assure public users the experts will safely serve their needs. The efficiency of smart contracts in removing middlemen and the open market of competing expertise tags will guarantee the most economical service.*

In a similar way collaborations for any purpose can be formed on the platform, by creating evidence of their expertise on the forum, collecting experts and vested employees in the bench, establishing reputation through the betting pool, and collecting fees through smart contracts outside the platform.

In the next section we illustrate how the platform can improve any type of business by facilitating fair and efficient dispute resolution.

## 7.2 Example: Smart Contract Dispute Resolution

In this section, we detail an important application of the platform, dispute resolution in the distributed and anonymized environment of the crypto economy. We call this application of the platform **Distributed Jurisdiction (DJ)**. DJ illustrates how several expertise tags in the forum can work in parallel, as smart-contracting parties will typically require several skills from their arbiters, as detailed below.

Bench experts in the dispute resolution expertise tags will be called **arbiters** in this section.

To use the Distributed Jurisdiction platform, public users must choose their own smart contracting language. Therefore, the protocols of dispute resolution are entirely determined by the contract parties as stipulated by the smart contract at the point of creation.

Successful smart contract language is expected to be developed in the forum through the evolutionary process driven by the validation pool. The protocols suggested in this section are therefore entirely hypothetical.

## 7.2.1 Contract creation

When a smart contract is created, the means of dispute resolution is included before parties sign.

### 7.2.1.1 Break clause

A break clause in the smart contract will transfer to a 3rd party arbiter the complete power to disburse the encumbered assets of the smart contract between the parties. The break clause is triggered whensoever either party chooses to initiate a dispute.

### 7.2.1.2 Third-party arbiter

Distributed Jurisdiction is designated as the platform for dispute resolution of smart contracts by including a 3rd party arbiter in the respective smart contract—an open position for a randomly selected expert from the bench whose power is triggered when a contracting party initiates a dispute. The arbiter's power to disburse the assets of the contract between the other parties, once triggered, must be preeminent in order to ensure certainty of outcomes.

Before being selected to adjudicate any case, each arbiter posts an availability stake. This is a number of their sem tokens that will be included as the arbiter's upvote bet on their final post justifying their eventual decision on the case.

An arbiter is randomly selected, weighted according to availability stakes posted and according to the expertise tags specified by the contract. These tags decide values assumptions for the contract. Expertise tags signal the type of dispute and indicate the experience required of potential arbiters. Examples of expertise tags include relevant areas of law (property, employment, Delaware corporate law, etc.), technical matters disputed, languages, territorial or cultural assumptions, and anonymity level.

### 7.2.1.3 Fees

Fees are determined by market factors at any given point in time. Fees are calculated based on the availability of relevant arbiter availability stakes. Arbiters specify the fees they will accept for each sem token staked, the availability stakes, by advertising in the forum. The platform does not dictate the fees.

If the fees chosen by the contracting parties are insufficient, the arbitration will fail as the arbiter selected will refuse to do the work. If the arbiters require fees higher than the public is willing to pay, they will fail to attract cases.

The smart contract must specify that the fee is sent in the arbiter's name to the Distributed Jurisdiction platform associated to the arbiter's judgment post in the forum. The fee is not sent directly to the arbiter, or else the parties are not technically using the Distributed Jurisdiction platform.

The reason the parties will prefer to send fees to the platform instead of the arbiters directly is the insurance provided by the evidence-of-work post, which only rewards the arbiter if other arbiters approve of their decisions.

### 7.2.2 Arbiter chosen

A random arbiter is chosen based on availability, expertise, reputation, and fees. Before the choice is made, all arbiters from the pool have submitted a stake signaling their availability, and the forum has advertised protocols and fees the arbiters will accept for dispute resolution work. The arbiter's availability stake is a number of their sem tokens, chosen from each of the expertise tags they've improved.

Then the system takes the weights that the contracting parties chose for each expertise tag (summing to 100%) and multiplies these weights by the available users' weights to determine the likelihood of each arbiter being picked. Using this distribution, the arbiter is randomly chosen from the pool of arbiters.

As an example, imagine there are 3 expertise tags in the contract, which the parties chose as 50% English speaking arbiter, 25% intellectual property law, 25% corporate law expertise. Next suppose arbiter A has 10% of the available sem tokens staked from the entire bench in English, 5% of the intellectual property, and 7% of corporate law expertises. Then $(0.5)(0.1) + (0.25)(0.05) + (0.25)(0.07) = 0.08$ so arbiter A has an 8% chance of being selected for the dispute by the system.

Once chosen, the arbiter has the power of disbursing the assets of the contracts as seen fit between the disputants. Asymmetry in party asset encumbrance may be abused if assets from each party are encumbered equally. However, the arbiter also has the power to combat such abuses by choosing at any point to release a portion of the property to either party before the judgment is complete, based on temporal concerns.

The arbiter's stake also represents a bet that the community will agree with their decision. An arbiter's stake is combined with the half the contract fee as the arbiter's upvote bet on the post of the decision, half the fee is staked as a downvote against the bet. This encourages careful participation of arbiters in the adjudication portion of the platform.

### 7.2.3 Private hearing

A private hearing is initiated with the arbiter as moderator. This hearing takes place entirely off platform, following the protocols set by the smart contract. Solutions for how to perform the hearing will be decided as protocols evolve in the forum.

One possible approach for the beginning of the platform is detailed as follows. Symmetric encryption (PGP, e.g.) guarantees efficient, anonymous communication between the parties and arbiter using their asymmetric public-key identifiers. Then the trial proceeds as follows:

1) Plaintiffs submit their case, including proposed settlement.

2) Defendants submit their rebuttal, including proposed settlement.

3) Arbiters ask open questions until deciding to close the question session.

4) Arbiters repeat 1)-3) until choosing to close the trial.

Standard procedure small asset contracts may have an arbiter ask no questions and deliver a verdict after step 2 by quickly explaining the relevant precedent. Standard procedure for larger asset contracts may involve several rounds.

a) 7.2.3.1 Judgment posted

The decision of the arbiter is posted to the forum, with official spaces for comments by the claimants. The arbiter can only benefit by increased reputation if the post wins in the upvote validation pool.

### 7.2.4 Justice mining

The posted judgment is viewable by the pool of arbiters in perpetuity in the forum. Comments praising and criticizing judgments will serve as a record of precedence in the system. Comments live as records of votes eternally, but the system allows for review of comments by reposting the same or similar comments further down the branch for a new, potentially more decisive vote, allowing further confirmation of a precedent or the possibility of overturning it.

The system employed to encourage careful adjudication is the betting pool. Arbiters stake a percentage of their reputation on an up- or downvote. After 1 day[8] of voting the results are revealed and the winners of the decision (upvoters or downvoters) split the losers' stakes, weighted according to their bets. This method determines justice with a democracy weighted according to expertise and encourages effort in discernment of truth.

There is an obvious threat to the anonymity of parties and arbiters in any adjudication process which requires information to be exposed during arbitration and the judgment post. This threat is minimized by good practices in adjudication, such as the use of highly standardized template language which will develop in the forum.

### 7.2.5 Appeals

The rules of the appeals process is encoded in each individual smart contract. Therefore the specifics of the process will evolve as best practices are discovered in the forum and upvoted. In this section we detail one possible scenario.

If an appeal is triggered by a claimant within the designated statute of limitations, a second hearing is initiated. A much higher stake is posted by the appellant to restrain frivolous abuse of resources. A panel of 3 arbiters is randomly selected, and power to disburse the assets is transferred from the

[8] The precise time period would ultimately depend on the area of contract law, determined by the bench. See §9.1 Parameter maintenance.

original arbiter. In this case the only variation in the hearing is to repeat the question phase, Step 3), with the 3 arbiters asking questions, in order, according to reputation.

When a majority of arbiters choose to close the hearing, a private conversation between arbiters is initiated to settle the dispute. When a majority of arbiters choose to close the private conversation between arbiters, a binary vote between claimants is cast. The arbiter with the greatest reputation in the majority opinion writes and posts the judgment, with space available for dissenting opinions.

A second and final appeal is possible, within the contractually specified statute of limitations, which repeats the process with 9 randomly chosen arbiters and a higher stake posted by the appellant.

As stressed previously, ultimately the type of appeals process is always determined by the parties of the smart contract at the point of signing. As the Distributed Jurisdiction platform matures, standards for statutes of limitations will evolve determined by the jurisdiction related to the relevant expertise tags. Similarly, the very process of appeals may vary depending on subject matter. For example, large disputes with less concerns for privacy, such as DAO forks, may begin with several arbiters who are not anonymous. In some cases it is possible arbiters are not even randomly chosen, but are respected figures in the field specified during contract creation. In such cases a dispute may be coded to begin when a quorum percentage of members triggers the break.

Appeals are a challenging element of the Distributed Jurisdiction system, since contradictory demands need to be satisfied. Due to the potential anonymity of claimants, the assets of the contract must be bound by the first smart contract until the end of all possible appeals, which puts the goals of careful deliberation and timely resolution at odds. The crowd-sourced efficiency of the open comment system as an experimental proving ground for good practices is again invoked to give solutions for this challenge.

### 7.2.6 Triage

The details of smart contract creation given above place a significant intellectual burden on the parties—they need to be experts in which type of contract to choose and then which type of dispute resolution protocol is appropriate including weights of expertise tags in various areas of law. It is assumed that the forum will solve many such problems, creating standardized contract templates that are easy to choose for the vast majority of business needs. However, there will always be novel situations, and in order to make the platform more attractive to business use, another layer will be developed which adds a 4th party triage expert who can make such decisions after a dispute starts.

Under this more complicated system, contracting parties can simply add a randomly selected triage expert to their smart contract. Triage expertise will be a separate expertise tag in the forum. A triage expert will have reputation in the expertise of deciding the areas of case law in which general contracts fall. The smart contract gives this triage expert the ability to transfer part of the adjudication fee and the power of asset disbursement to the appropriate expertise tags to randomly select an appropriate arbiter. Then following the general platform function, the triage expert will submit their work for evaluation by other triage experts to the forum.

### 7.2.7 Attacks

In addition to the general attacks listed in §4.1 and §8, this dispute resolution example is susceptible to another type of corruption: arbiters may be bribed. When both arbiters and parties are anonymous and the channels of communication are carefully limited, this is far less likely than in legacy legal systems.

However, when the bench is limited or when anonymity is neglected, the danger may seem more pronounced than it is in the traditional court system. For instance, an anonymous judge has the opportunity to blackmail a known claimant. Or when an arbiter's pseudonymous identity is revealed, a claimant may attempt a bribe. Or when the bench is small, a powerful private user could spend a great deal of money engaging the bench with separate contracts beforehand, as happens today in the U.S. when traditional judges are given regular consulting fees by powerful firms to ensure preferential treatment in the future.

For such situations, the system of checks and balances greatly inhibits the potential for corruption. Arbiters are highly incentivized to maintain probity, since their evidence-of-work post will be policed by the entire bench of arbiters with the threat of loss of availability stake. If that is not enough to dissuade corruption, the potential of diminishing the value of all of their reputation is high if they manage to hurt the reputation of the expertise tag.

Further, smart contracts written as suggested above give insurance an unfair settlement will never occur if 1) anonymous arbiters are randomly chosen according to the weight of their availability stakes, 2) arbiters are required to post evidence-of-work posts for evaluation by the forum, 3) claimants are given automatic space in the post for comments, and 4) a system of appeals is included. In this case, the policing of unfair adjudication is automatic, and the existence of appeals severely weakens the power of bribery.

### 7.2.8 Freedom of Contract

Distributed Jurisdiction epitomizes the principles associated with the freedom of contract. Smart contracting parties may choose contractual terms to incentivize contract formation or protect themselves from unwanted outcomes, especially in an anonymous contracting environment. Such choices may not always be the most economical but may protect parties to smart contracts. Below we list some forms of contractual terms that parties to smart contracts may consider.

#### 7.2.8.1 Symmetric encumbrances

Once a break clause is triggered, the contract in question is suspended in its current state. In the crypto economy, where the anonymity of one or both parties is to be expected, the assets in dispute (money, property, reputation, etc.) need to be bound in the smart contract. In the blockchain environment it is impractical to seek assets from an anonymous party who may have transferred the assets to new anonymous parties during the interim. Further, it would be damaging to the business platform to make the attempt to retrieve assets of a dispute from outside anonymous parties further down the transaction tree.

Therefore, it is incumbent upon the parties not to enter into an asymmetric contract of bound assets, lest one party may unfairly exert power over the other by encumbering assets in a dispute.

For a party to a smart contract that provides services, for instance, symmetric encumbrance may mean that the service provider stakes sem tokens. Thus, for simple contracts, the Distributed Jurisdiction may be seen as an efficient solution for bringing an escrow service to transactions, small and large.

#### 7.2.8.2 Plaintiff stake

To discourage abuse of the remediation process, it is expected (but not necessary) that a plaintiff will add a fee to the contract, called a plaintiff stake, in order to trigger the break clause. In the event of a frivolous dispute, the plaintiff's stake will be disbursed by the arbiter between the defendant and the Distributed Jurisdiction platform. The size of the stake will be specified by the smart contract, determined previously by the parties.

#### 7.2.8.3 Human language

In order to use a human arbiter, it is expected (but not necessary) that many contracting parties will attach a human language version of the contract to clarify intent between parties and for the arbiter's use in the case of a dispute, as in a Ricardian contract.

## 7.3 Example: Company organization

Any type of collaboration may be organized through the architecture in order to vest users with work-verified reputation. As an example consider the following hypothetical development of a company within the platform, which has important ramifications for the evolution of the concept of the firm..

Adam has an idea to create a product, and creates an expertise tag, E1, with an arbitrary amount of currency, say for example, 10 units. Automatically, 10 sem tokens of type XRP-E1 are minted and added to Adam's new expert account (see §3.2.2 New expertises). Adam, as founder and sole owner, has complete power as CEO over the evolution of the E1 expertise tag/company, until choosing to relinquish control to trusted experts who have proven their value, or by selling his sem tokens.

In order to attract outside talent and resources, Adam posts to the E1 expertise tag of the forum his plans for the company's structure and future development and policies for joining. Public users apply for positions within the company/expertise tag by posting to the E1 tag, following Adam's posted protocols, e.g., by staking currency less than 1/10th Adam's original stake, leaving Adam controlling share of the company. Adam evaluates and accepts or rejects each post.

The accepted public applicants are vested with power in the company according to their holdings of sem tokens granted by Adam. These new experts may now influence the direction of the company/expertise tag by posting evidence of work and evaluating other's work. Adam retains control by using his outsize share to enforce protocols which protect his majority.

Through work verified in the forum by the validation pool, the company eventually increases its value until it can credibly promise a product to the public in the E1 expertise tag of the forum.

E1-approved smart contracts are engaged to deliver the product. These contracts specify product details, fees, and the particular experts which will provide the product. To collect the fees, evidence of delivery is posted to the forum, with comments from the public party, for evaluation by the company experts. This system automatically provides an open record of business that can be evaluated by the public to determine the trustworthiness of the company.

Share sales in the company can be announced and organized in many ways through the forum. One approach is for public investors to post stakes to the forum as empty comments. Those who are accepted (presumably because they didn't try to purchase too many tokens) gain sem tokens, which vests them with the rights to a (variable) fraction of company decision-making power and all future fees the expertise tag/company collects. The ability of the experts to change the exchange rate between off-platform currency fees and company sem tokens (as explained in §9.1) makes stock creation easy.

Creating a company is very efficient on the platform. The open record of its entire history means it is easy to copy the structure and organization of successful companies. See Appendix §8.1, however, for a discussion of the danger this poses in the construction of sham companies, and how to counteract it.

## 7.4 Example: Machine experts & formal verification

A key design goal of the platform is to create a minimal and robust architecture that allows for creating much more complex and automated validation logic, as well as machine expert validators.

For instance, experts could evolve from humans to smart contracts in order to allow for the automation of validation. In addition, the forum can support the creation of formal verification proofs and engines, which could be utilized by experts in their validation process. In this section we discuss proof of stake, oracles, and machine arbiters.

### 7.4.1 Example: Proof of stake

Blockchain theorists are attempting to improve on the proof-of-work protocols initially deployed on the Bitcoin platform, in order to decrease the energy it takes to cryptographically secure transactions, and scale the system so that it can handle a higher rate of transactions[9]. The platform provides one means of improving so-called proof-of-stake (PoS) protocols.

Instead of solving the complicated proof-of-work hash puzzles, **proof-of-stake** protocols dictate that a lottery is held amongst users with verified reputation. The lottery determines who gets to add the next block in the blockchain, claiming the fee reward.

[9] See, e.g., https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ

Using the platform, experts will stake their reputation to determine their weighted likelihood[10] of being selected in the lottery, an availability stake, as detailed in §3.2.4. Instead of the lottery winner collecting the fees from the block of transactions, the fees will be paid in reputational salary to all verified experts. Then newly minted sem tokens of the same quantity as the fees are staked ½ for the lottery winner and ½ against in the validation pool. This gives the rest of the experts the opportunity to evaluate the validity of the lottery winner's block. The lottery winner is automatically rewarded with the majority of the new sem tokens if the block is valid. The rest of the experts are rewarded with smaller shares of the newly minted reputation for policing the lottery winner.

Blocks will only be accepted by the network if they have proof of success in a betting pool. This stamp is the entire cryptographic security, which costs far less energy to achieve than proof-of-work token mining.

There are many choices possible for deciding the process for accepting new experts. For simplicity, let us first consider the stability of the least complicated option. Let us assume the selection process is left open to anyone who wishes to pay any fee (§3.2.1 New experts), we claim the system is difficult to corrupt by the following reasoning. The cost to acquire 51% of the sem tokens by paying fees to the system was calculated in §4.1.4. In the worst-case scenario, when anyone can add any number of fees and be vested immediately, without review, and no other users are adding any fees beside the malicious users, it will cost much more than twice the total number of sem tokens, $g_0$. If the malicious group is merely doubling the rest of the world-wide investment in continuous fee payments, then it will take more than 6 times as much as the global historical fees of the system. If the malicious group attempts larger lump payments instead of continuous micro-investments, it will require a far higher outlay.

There are two reasons a malicious group might wish to profit from the system by making a flawed block, either to profit by stealing the fees from the particular block or to destroy trust in the entire blockchain.

Assuming the malicious group does manage to take over the platform, the rest of the world would immediately see the unfair action that couldn't be rectified by the good-faith actors. So the platform would topple. The malicious group would merely gain the (fraction) of the fee from one block, while losing their investment. As long as any single block's fees are smaller than twice the total reputation, there is no incentive to attempt to game the system for fees.

The only other reason to corrupt a block is to destroy faith in the platform. But then the $>>2g_0$ invested becomes worthless. So the cost at any time to topple the system is more than twice the total sem tokens—assuming you can simply buy reputation without any temporal limit.

[10] Creating a pseudorandom number generator in this context is a little complicated. The seed for the generator that determines the next block author is partially controlled by the current block author, which opens the possibility of gaming the system by controlling all block creation by sending authorship to your own Sybil accounts. To combat this attack, we impose the following protocol: The seed for the number generator will be determined by a hash of the information given during the validation of the previous block, such as the symmetric keys of all validators. Cf. step 10 on page 9 of the typical flow of events for a user interacting with the core.

Therefore as long as the fees are low compared to the reputation total, or if there is no competitor that is suffering by the platform's existence by more than twice the total number of sem tokens, there is no incentive to corrupt the system—it is far more valuable to use such power to improve the platform.

More importantly, everyone besides the malicious actor is rewarded with the malicious actor's fees. So the malicious actor cannot really hurt the creators of the platform. The malicious actor can only harm their own investment, while helping the others.

As fees are added, reputation is added, so the system becomes more secure as it is used.

Even if there are many different malicious actors in the system, if no single malicious group gains 51% control, they will never be able to corrupt a block to gain the fees.

There is another way to acquire tokens, by merely buying any available on the open market, which we address next. Under this scenario Houy's criticism[11] of all PoS protocols is as follows: If a credibly rich person wishes to destroy the blockchain, it will cost them nothing, because merely by signaling their intention to destroy it by spending whatever is necessary to purchase 51% of the staking tokens, the tokens will devaluate to 0, since holders will enter a race to the bottom to divest themselves of a token that will soon be worthless. However, in the current example, the value of the token is not merely tied to public opinion. The value of the token is calculably predictable. And when someone signals their intention to buy tokens—for whatever reason—the price usually goes up, not down.

Therefore the price to purchase 51% of the tokens on the market will be quite high. Particularly because most tokens will be in constant use since the automatically make money for their users, so there will be a scarcity of tokens available through this channel. (See §4.1.3 The 51% attack.)

Finally the system becomes far more secure if any other safeguards are instituted, such as triggering appeals, as discussed in §7.2.5 Appeals.

### 7.4.2 Example: Oracles

An oracle for a blockchain is a system that allows smart contracts to access real-world (i.e., off-chain) data. Since the blockchain is an autonomous system with no centralized authority dictating decisions, there is a need for a trusted 3rd party to reputably state the results of real-world events. For instance, when blockchain users wish to exchange a smart contract which mimics a stock option, they need agreement on the value of the stock at the expiration date, which generally requires a trusted 3rd party. For the success of the crypto economy, the 3rd party should remain decentralized and anonymous. The platform gives such 3rd parties an opportunity to establish verified and valuable reputation.

There are many uses for oracles on an autonomous distributed system, and there are many approaches to creating oracles on the platform. One simple approach is to create several oracle expertise tags, consisting of expertises devoted to deciding the outcomes of real-world events in

[11] Houy, Nicolas, It Will Cost You Nothing to 'Kill' a Proof-of-Stake Crypto-Currency (January 2014). Available at SSRN: https://ssrn.com/abstract=2393940 or http://dx.doi.org/10.2139/ssrn.2393940

various subjects, such as sports, political races, stock prices, weather, etc. The forum hosts the posts of reported outcomes, the veracity of which is decided by the validation pool.

Smart contracts can be written to choose the duration of the validation pool, within some limits decided by the bench experts in the particular expertise tag relevant to the subject matter being validated (see §9.1). For instance, sports bets may have a time scale in days, whereas weather reporting may be automated and occur on the scale of minutes. The validation pool may conclude before the event occurs to enable speculative pools, or after the event occurs to allow bench review of a judgment about the past. The core will be written to allow the option of validators to reveal their bets if a particular expertise tag allows it.

Smart contracts developed in the forum, would call bench experts (as stated before, we encourage random selection weighted by reputation) to write posts in the forum deciding the truth of an event, which would then be evaluated for veracity by the rest of the bench through the betting pool as described previously. Specifically, the smart contract gives the chosen bench expert the power to send the fee in the expert's name to the platform, which then mints sem tokens, stakes half in the chosen experts name for the bet and half in the bench's names against. The validation pool is resolved and the tokens are distributed, then the fees go to salaries for the bench weighted by reputation.

For complicated and messy real-world outcomes, human judgment may be necessary. But for many oracle needs, such as reporting stock prices, the entire system could be automated and therefore would require smaller fees to achieve stability.

### 7.4.3 Example: Machine arbiters

Consider the example of a decentralized system which facilitates the cooperation between two machines, Alice and Bob. The goal is to complete a fair exchange between two parties, i.e., a transaction where Alice and Bob transfer files between each other per previous agreement. A classical problem is inventing a protocol which guarantees both parties fulfill their promises by including Ted, a third party (machine) arbiter which gains control of the files if either party signals dissatisfaction.

Under certain restrictive assumptions, it has been proven that no protocol can be created which will guarantee fair exchange with decentralized arbiters[12]. Despite this lack of absolute certainty, the architecture provides a means for developing a platform which can give users confidence that business can proceed through a system of verified reputation. This gives a calculable probability of confidence, even if no provably certain algorithm exists.

By creating a "Fair Exchange Machine Arbitration" expertise tag in the forum, experts can gain reputation by 1) providing effective smart contracts allowing the public to use the expertise tag for arbitration, 2) fairly resolving automated disputes, and 3) posting evidence to advertise the insurance the tag provides. Each time an expert is used, the bench experts all benefit, which motivates the healthy development of the expertise tag. Each time an arbiter gives an unfair

[12] Küpçü A., Lysyanskaya A., "Optimistic Fair Exchange with Multiple Arbiters", In: Gritzalis D., Preneel B., Theoharidou M. (eds) Computer Security – ESORICS 2010. Lecture Notes in Computer Science, vol 6345, Springer, 2010.

judgment, appeals have the opportunity to correct the decision, so the offending arbiter would likely be punished with loss of stake and blacklisting.

There is no absolute certainty using the expertise tag will result in a fair resolution. Malicious experts can always choose to use an unfair algorithm to distribute the assets. In that case, however, appeals and blacklisting prevent them from gaining immediate rewards and future power. Thus, a probabilistic confidence in fair resolution will evolve that will increase with the value of the reputation tag.

The number of fair decisions and honest arbiters will be statistics available for public analysis through the open record of historical performance. The rate of fees paid to the expertise tag through public use will be statistics collectable from the forum, so the value of sem tokens in the expertise tag will be open to public analysis. Therefore, each transaction value will have a numerically verifiable confidence level at which the expertise tag can guarantee fair arbitration.

Through increased use, the transaction values the expertise tag can handle will continually increase, along with their confidence level.

## 7.5 Example: Policing crime

As a final example, we discuss an issue that is important for the general acceptance of the crypto economy. The goal of promoting freedom is well established in the crypto community. But this is often used to criticize the motives of the crypto community, as they are regularly characterized as promoting lawless behavior. A system that allows anonymous users the ability to efficiently organize and cooperate in any endeavor will be used by all sectors of the economy, and inevitably it will be used for many types of unethical, malicious, and criminal activity.

However, the majority of contributors to the crypto community have largely altruistic motives. Anonymity and decentralized trust also permits much more effective methods of policing crime and corruption. For the first time there is an efficient means of encouraging the reporting of crime. A crime-witness expertise tag can be created which gives public users a safe opportunity to anonymously report crime, and a means to insure they will be remunerated. One way to organize the expertise tag is as follows.

First, the crime-witness forum would specify the protocols and smart contracts that would be used by witnesses. These protocols would include the opportunity for any public user to become an expert witness with a very small fee set as a maximum allowable. Then open bounties would be set by the community with public smart contracts sending fees to the platform waiting for expert witnesses to step forward.

When an expert witness accepts the bounty contract, they are given the opportunity to post their evidence to the forum with the public bounty as the fee in their name along with their expert witness sem tokens. The rest of the experts from the bench will evaluate the witness's post to decide if the evidence is sufficient. If so, the witness will gain the sem tokens minted for the post by the platform, not the fee directly. The witness will be paid in future reputational salaries.

If the evidence is not sufficient, the bench will vote the post down in the validation pool and the witness will lose all their tokens. The smart contract can be written so that the total bounty is sent to the platform in the poster's name if the evidence is sufficient, but retained by the contract if it is voted insufficient. One way to do this is to code 2 potential rounds of validation in the bounty smart contract. The first one would be associated with a minimal fee; the second one would occur for the complete bounty if the first one resolved in the witness's favor.

Witnesses can be confident their anonymity is protected and their actions will be properly remunerated due to the evidence of effective posts, historically recorded in the forum.

# 8 Appendix: Critical analysis

In any open system it is important to identify security weaknesses. Particularly challenging aspects of any reputational system with anonymous accounts is the ability to influence the platform by purchasing sem tokens for Sybil accounts, tyranny of the majority, and the 51% attack, as discussed above in §4.1.1-3. We address further suspected points of failure in this section.

## 8.1 Sham expertise tags

Part of the power and efficiency of the platform includes the ability to easily create expertise tags. The open forum gives a record of work and business for any successful expertise tag, which can easily be copied and reposted under a new tag name for the low cost of the anti-DoS fees. The capital used to mimic fees paid to the sham expertise tag is mostly recovered by the sham owners through the salaries they control.

In the open, decentralized, anonymous environment of the platoform, we provide no mechanism to stop the formation of such sham expertise tags.

How, then, is the public able to decide which expertise tag is reputable? In the same way web browsers and search engines steer us away from untrustworthy clones of websites, the open record of the history of similar reputation tags should give the public the power it needs to make an informed choice. E.g., precedence and longevity will be easily comparable, between reputable and sham expertise tags.

## 8.2 Accumulation of reputational power

The inevitable accumulation of sem tokens by anonymous users is another potential source of corruption. Reputational power can be sold via the sale of anonymous identities. Such corruption is checked by the existence of 51% good-faith users, who will vote fairly in the validation pools. If 51% of users collude to maliciously enrich themselves, nothing can prevent them, except the open system itself, which would quickly detect such an attack. This revelation would gut the value of the system in which the attackers have a majority holding, since public users would no longer send the expertise tag fees, so the strategy is ultimately counter to their interests.

Secondly, as the system reaches a semblance of maturity, the power of experienced experts will far outstrip that of new experts. Especially in a system which requires staking reputation in order to adjudicate disputes, how are new experts able to earn reputation? New experts may post comments using stakes with currency. If their comments are accepted by the community, they will gain reputation, since more than half their currency stake is returned to them as sem tokens. But the review system prevents malicious users from simply buying sem tokens, since comments that are not perceived as improving the platform lose their stake.

The accumulation of power in the platform is naturally mitigated by the constant influx of reputation. As disputes are settled and fees are collected, the fees are continually converted into new sem tokens. Since the system splits wages equally among users relative to total system reputation, older sem tokens constantly lose power as time goes by. Therefore, powerful users can only maintain their positions by adding useful work to the platform. See §5.4.2.2 on reputation calculations.

If experts in a bench identify the problem that not enough new experts are joining the expertise to properly serve the public smart contracts they may choose to attract new experts by adjusting the exchange rate between ETH fees and sem tokens via the mechanism described in §9.1 Expert parameter maintenance.

Finally, any expertise tag which is corrupted will not attract outside fees, and it is easy to create a competing expertise tag to replace it. Thus there is no incentive to deliberately harm any expertise tag in this way, because it takes money and effort to gain power in the expertise tag, you can only harm the system with the power you have, it is easy to police this open system, and harming the platform harms your investment.

## 8.3 DoS attacks

Spam or DoS attacks are inhibited as usual with nominal fees for new posts as determined by Ethereum.

# 9 Appendix: Ancillary functions

In this section we detail further complications which are not crucial to the basic understanding of the platform, but will give extra power to experts as they use the system by their inclusion in the core: 1) the ability to adjust basic parameters in the core, such as the expiration timing for validation pools, 2) the ability of users to weight the power of precedent posts in the forum, and 3) the ability of users to request a validation pool for half price without staking tokens on the upvote.

## 9.1 Parameter maintenance

To maximize efficiency, several parameters within each expertise must be regularly adjusted as the platform evolves. Different expertise tags will have different values for deciding these parameters. For example:

- the expiration time for a validation pool

- conversion rate between off-platform currency fees and sem tokens
- the minimum stake/fee for comments
- whether the validators' votes are revealed or hidden before the validation pool expires.

To promote autonomy, core programming of the platform will include a function which allows separate validation pools for the purpose of allowing experts to adjusting these parameters within each expertise separately. The forum will have branches devoted to program improvement. The upvote system will provide ample decision-making mechanisms for debate and resolution.

Regarding the time limit for the validation pool, first, each validation pool should conclude early if all active experts have voted. An active expert is defined as an expert who has voted in one of the last 5 consecutive active pools. An inactive user becomes active as soon as they vote in any pool. To add stability, only one parameter change proposal can be active at any moment, and each proposal would have lower and upper bounds of parameter adjustment limited to halving or doubling their current state.

A more sophisticated mechanism, useful for example in an oracle expertise tag (as detailed in *§8.5 Example: Oracles*), would allow smart contracts to specify the duration of the validation pool within a choice of ranges decided by the bench experts.

## 9.2 The forum as a weighted DAG

If the tokens that are minted for each post have equal weight, then comments which are not controversial are not rewarded. A comment which makes a clear improvement to the platform, assuming it is extremely popular and universally upvoted, will not be directly rewarded since there would be no contrarian reputation staked and lost.

Similarly, successful comments (assuming very low DoS fees) give their posters no greater reward than their fellow upvoters receive. So the same reward is given for crafting a comment as is given for reading and voting, which encourages voting over commenting.

We might attempt to justify this format by arguing universally popular opinions are obvious, or that it does help creators by improving the expertise in which they hold tokens. But that would be disingenuous. This is a clear flaw in the system which needs to be improved by weighting the posts.

Weighting the value of tokens—relative to how they are received by the bench in the betting pool and by how they are referenced by other posts—incentivizes these universally well-regarded contributions. For instance if there is a large branch of positive references based at a node, the tokens minted at that node should be worth more. If the post was contentious, and its betting pool was close to 50-50, the tokens will be worth less in future salaries than uniform agreement. The amount of interest generated by the post will be measured by the percentage of the platform that votes on it.

However, all these suggestions are subjective value-judgements, and there are arguments to be made for implementing their opposites.

With this added consideration, the graph structure of an expertise in the forum becomes a citation[13] graph, a type of a directed acyclic graph (DAG), instead of a simple tree graph, since posters may wish to reward or punish more than one previous post, with their reference. The platform allows users to specify their own weight for the citation on a continuum from -1/2 to 1/2, where 1/2 means their post is in agreement with the citation and -1/2 means their post is opposed to the citation. Thus citations are the weighted edges of the forum graph. Then the tokens minted in each post would have separate values based on their relative position in the weighted DAG. Next we detail a simple method for valuating these tokens.

Denote the posts by $p_n$ numbered chronologically for $n = 1,2,\dots,N$ where $p_N$ is the newest post. Denote the following symbols based on $p_n$

- $w_{n,k}$:= "the weight of a reference from $p_k$ to $p_n$". So we have $-1/2 \leq w_{n,k} \leq 1/2$with $\sum_k \quad |w_{n,k}| \leq 1$ for each $n$.
- $u_n$: ="number of upvotes on $p_n$".
- $d_n$: ="number of downvotes on $p_n$".
- $a_n$: ="number of active voters during the pool for $p_n$".
- $v_n$: ="value of a token created for $p_n$ initially": $= R(u_n - d_n)/a_n$ where $R(x) := max\{0, x\}$ is the ramp function.
- $c_n$: ="number of tokens created for $p_n$"
- $t_n$: ="total value of $p_n$": $= R\left(v_n c_n + \sum_k \quad w_{n,k} \cdot t_k\right)$

Then the value of 1 coin minted for post $p_n$ is $t_n/c_n$ and the reputation-weighted salaries will be paid for each token according to its relative weight within the expertise.

With this choice of valuating $p_n$ according to the recursive formula given for $t_n$ we are specifying that

1) posts which were initially controversial are initially worth less according to the term $v_n c_n$ since in that case$v_n \approx 0$
2) references add value according to their weight via the term $\sum_k \quad w_{n,k} \cdot t_k$
3) the ramp function $R(x) := max\{0, x\}$ guarantees no post can be devalued below 0, since tokens cannot have negative value

[13] The name comes from the field of bibliometrics where researchers study the importance of academic and legal documents based on the number of citations they receive. Citation analysis is the subject which studies the patterns and statistics of citation graphs. Traditionally, however, scholarly documents do not explicitly weight the power of their references.

4) references $p_n$ with $w_{n,k} = 1/2$ gain relative value equal to the post $p_k$ making the reference; chains of such references multiply the effect, opening a security risk, which is why we restricted it to $1/2 < 1$

5) if a post $p_n$ changes value by being newly referenced, all posts that $p_n$ references will also change value; so the value of every post is eternally dynamic; this gives the ability to update the power of precedents as the expertise changes

6) Unfortunately a post $p_n$ which was initially downvoted can never give its creator tokens, even if opinion eventually reverses. This could be changed if we decide to alter the core programming of the platform to grant tokens to winners and losers of a post, and allow the value of those tokens to be negative based on their votes and previous references. In this case if a downvote on a post was disagreed with in the future, the negative tokens could eventually become positive. If Sybil accounts are a threat, negative token holdings should not decrease an owner's reputational salary.

Notice the matrix $[w_{n,k}]$is upper diagonal, since posts may only reference past posts. This makes the determination of $t_n$ much easier in the platform's DAG than in a more general network, e.g., Google's PageRank algorithm, which means the cost of running the algorithm on a distributed network such as Ethereum is not prohibitive.

## 9.3 Parallel tags

Many uses of the platform will require experts with reputation in a variety of expertises. In this case, the smart contracts can select experts by signaling their preferences with weights. Then the random selection will be adjusted by the weights, and the fee will be split between the expertises according to the weights. The separate expertises would have separate validation pools. Each pool could evaluate a single evidence-of-work post, or separate posts could be sent to the separate pools.

## 9.4 Half-price validation pools

As stated above, when a user, Bob, submits a post along with an ETH fee to the platform to initiate a validation pool, two distinct things happen:

1) Bob requests a validation pool.

2) Bob bets on the Upvote with half of the minted tokens, which are assigned to him

These are two distinct actions. For creating a more flexible system, it is better to split these two actions into two separate calls to the core platform: one to **ask** for something to be validated; another one to **bet** on something to be validated. So instead of bundling them automatically in a step, they would be decoupled.

This allows users to not send the half of the fee which is staked in Bob's name. In some cases users may wish to use the validation pool to poll experts to decide a question, even though the users are not interested in gaining reputation in the expertise.

## 9.5 Other networks

The architecture for the platform was created to be effective in a hostile open environment containing malicious anonymous actors. It is more effective in a centralized system with identified participants where the blockchain is not necessary and its inefficiencies can be avoided.

# 10 References

Vitalik Buterin, “Ethereum: A next-generation smart contract and decentralized application platform”, 2014. [Online] Available: https://github.com/ethereum/wiki/wiki/White-Paper

Luis Cuende & Jorge Izquierdo, “Aragon Network: A decentralized infrastructure for value exchange, Version 1.1”, April 20th, 2017, [Online] Available: https://aragon.one/network/

Richard Dennis & Gareth Owen, “Rep on the block : A next generation reputation system based on the blockchain”, The 10th International Conference for Internet Technology and Secured Transactions, 2015. [Online] Available: http://www.the-blockchain.com/docs/A%20next%20generation%20reputation%20system%20based%20on%20the%20blockchain.pdf

Robert C. Ellickson, Order without Law: How Neighbors Settle Disputes, Harvard University Press, 1994.

F. Randall Farmer & Bryce Glass, Building Web Reputation Systems, Yahoo Press, 2010.

Gnosis Whitepaper 05.04.2017, [Online] Available: https://gnosis.pm/resources/default/pdf/gnosis_whitepaper.pdf

Zackary Hess, Yanislav Malahov, Jack Pettersson, “Æternity blockchain: The trustless, decentralized and purely functional oracle machine” DRAFT, December 28, 2016. [Online] Available: https://aeternity.com/aeternity-blockchain-whitepaper.pdf

Küpçü A., Lysyanskaya A., “Optimistic Fair Exchange with Multiple Arbiters”, In: Gritzalis D., Preneel B., Theoharidou M. (eds) Computer Security – ESORICS 2010. Lecture Notes in Computer Science, vol 6345, Springer, 2010.

Manuel Sebastian Mariani, Mat´uˇs Medo, & Yi-Cheng Zhang, “Ranking nodes in growing networks: When PageRank fails”, Nature, Nov. 2015.

Satoshi Nakamoto, “Bitcoin: A peer-to-peer electronic cash system”, 2008. [Online] Available: https://bitcoin.org/bitcoin.pdf

Elinor Ostrom, Governing the Commons: The Evolution of Institutions for Collective Action, Cambridge University Press, 1990.

Jack Peterson & Joseph Krug, “Augur: a Decentralized, Open-Source Platform for Prediction Markets”, [Online] Available: https://bravenewcoin.com/assets/Whitepapers/Augur-A-Decentralized-Open-Source-Platform-for-Prediction-Markets.pdf

Alex Rea, Aron Fischer, Jack du Rose, “COLONY, Technical White Paper 20170920”, [Online] Available: https://colony.readme.io/docs

Alexis deTocqueville, *Democracy in America*, 1835. (Harvey Mansfield and Delba Winthrop, trans., ed., University of Chicago Press, 2000.)