**Introduction to DeLaw: An Opensource Decentralized Legal Arbitration Protocol Providing Trusted Oracles**

**GitHub Address:** <https://github.com/ethdelaw/Delaw.git> - **Reddit: r/DeLaw**

**Dev Contact:** [**ethdelaw@proton.me**](mailto:ethdelaw@proton.me)

**Dev Fund Eth or MATIC Address: 0x4f551d675ac1d68caf8a6327CfADed6f493d3B1d**  
**Abstract**

DeLaw introduces a decentralized legal arbitration protocol that resolves the current limitations of smart contracts by providing a trusted Oracle system for subjective data interpretation. This protocol fuses blockchain’s immutability with traditional legal mechanisms, offering a streamlined, cost-effective alternative to conventional dispute resolution. Designed to enhance Ethereum Virtual Machine (EVM) functionalities, DeLaw caters to a diverse range of applications, from customer service applications, gambling settlement to insurance claims and beyond. Stakeholders are held accountable through stake-based incentives, and arbitrator selection is transparent, promoting fairness. Planned expansions include broader contract varieties, incorporating decentralized identity and credentialing and a DAO for protocol governance. This white paper details the innovation and strategic framework behind the protocol, promising a more efficient and accessible system for legal arbitration in the digital economy.

**Summary**

Blockchain technology has provided revolutionary leaps in automating and enforcing agreements without intermediaries. However, their potential is currently constrained by the absence of reliable protocols for evaluating Oracles that handle real-world, subjective, and complex data. This limitation hinders the expansion of Ethereum Virtual Machine (EVM) applications into areas requiring nuanced judgment and verification of non-binary outcomes.

DeLaw addresses this critical gap by introducing a novel arbitration protocol that merges established legal principles with blockchain technology. At its core, DeLaw leverages stringent incentive structures designed to secure the network and ensure the reliability of Oracles. By doing so, DeLaw not only resolves the "trusted Oracle problem" but also broadens the scope of possible EVM use cases. Further, it is argued that Traditional Legal frameworks have become too cumbersome and costly for efficient marketplace use, this protocol hopes to streamline and minimize the use of traditional legal systems increasing the decentralization of contracts within society while also providing higher value to its users. This white paper outlines our approach to building a decentralized system where legal arbitration meets cryptographic security, providing a robust foundation for trust and transparency in executing complex contracts.

Each step of the arbitration flow involves increasing staked value, giving incentives for reasonable settlement, deter bad-actors from arbitrarily requesting arbitration, and holding arbitrators accountable.

**Use Cases**

Use cases for such a protocol are immediately obvious. There is no limit to the requirement for trusted oracles to validate subjective results and ensure smart contracts execute payouts accordingly. Here are a few use case examples that we hope the community will take advantage of our protocol and build services on top of.

1. Gambling: Determining outcomes for instance in specific event wagers such as third round knockout or the color of the Gatorade poured on a coach at the Super Bowl.   
  
2. Customer Service Applications for instance decentralized delivery services where disputes arise over product or meal was delivered as ordered.

3. Insurance applications, the protocol can be used to assess and verify claims objectively aiding determination of approval or rejection of funding based on terms stipulated in insurance policy. Enables fast and reliable judgements preventing bad actors in the insurance space.

4. Contract resolution. Useful in scenarios where contractors require assurance of payment upon completion of work while customers seek guarantee that work meets a certain standard of quality.

5. Real Estate Transactions. Deposits are regularly placed for purchase or construction contracts. These are regularly abused by contractors or real estate agents. Such a platform can validate the reasonableness for delays, physical inspection claims or incomplete construction.

6. Freelance Work Agreements: Clear system to adjudicate disputes over deliverables, timelines or payment, particularly when work cannot be easily quantified.

7. Supply Chain Management: Resolve disputes over quality, timing or condition of goods delivered upon multiple steps in complex supply chains.

8. Peer-Peer transactions, can provide framework for disputing peer-peer transactions such as loans or service exchanges particularly where traditional legal systems may be too costly or cumbersome.

9. Community Genius, we are sure that the community will find new and innovative ways to expand upon these use cases.

**Contract Outline**

A diagram of a process

Description automatically generated with medium confidence

This follows the following description:

1. Contract Initialization: A contract is established and funded between two parties.
2. Dispute: If a claim is disputed after the contract is executed, arbitration commences.
3. Arbitration Outcome:
4. If the arbitrator's decision is not challenged, the contract is paid out.
5. If the decision is challenged, an appeal process begins.
6. Appeal Resolution: The appeal ends with a decision that determines the winning and losing parties.
7. Post-Appeal Actions:
8. The winning party has their contract and stake paid out. Remaining contract value is reserved depending on the loser’s appeal decision.
9. The losing party can either accept the decision or challenge it further within DeLaw or through traditional legal channels (TradLaw). It is important to note, that individuals here are not contesting the original case, but rather the reasonableness of the Appeal Arbitrators’ decision.
10. If there is no challenge, the arbitrators are paid their fees and remaining contract is paid out.

\*It is important to note that Post-Appeal Actions justify the closing of this contract and the creation of a new contract for either DeLaw or TradLaw to render the decision. In the case of TradLaw, the created DeLaw contract is created solely to assess the outcome of the TradLaw case.

**Contract Utility Functions**

Our initial protocol shall include these specific utility functions at launch:

1. Case Creation: Create case between two parties enabling contract establishment.
2. Provide Contract Deposits: Track funding of contract which is required before contract is valid.
3. Provide Contract Gas: Enables individuals to ensure sufficient native tokens in contract for gas fees, unused gas returnable upon contract completion.
4. ERC-721 Token assignment: This protocol uses ERC-721 Tokens as proof of contract validity. Contracts, dispute arguments and arbitrator’s decisions are uploaded to IPFS as an image or PDF ensuring non-tampered proof of details for future review. Arbitrators are granted read-access to these contracts to make their own determination and judgement. Future utility will a multi-sig solution to enable anonymous contract details to be viewed only by the applicable parties and arbitrators selected.
5. Staking Protocol: Enable efficient staking to fund contracts, staking is a requirement to claim settlement or dispute claim. Additionally, matching stakes must be provided from arbitrators to secure and enforce good decision-making principles.
6. Arbitrator Selection: Arbitrators can select a case by staking their decision guarantee. Staking can be conducted before reviewing case details to prevent arbitrators reviewing contract details for cases which another arbitrator is already reviewing. Future utility will include identity tokens to further secure the protocol with proof of identity and proof of qualification for arbitration selection.
7. Raise Dispute: Enable eligible parties to raise dispute by signing a message and staking required dispute fee to the contract.
8. Negotiated Settlement: Enable parties to agree to negotiated settlement terms during dispute process.
9. Appeal contested: Enable function for the appeal to be contested by either Traditional Legal frameworks or another round of DeLaw arbitration. Required staking value to challenge appeal is higher for TradLaw to compensate for additional cost burden associated with disputing claim in court (jurisdiction as required by contract).
10. Contract Time Extension: Before the contract expiry date, both parties can agree to extending the contract completion period for no cost.
11. No Challenger: If no challenge exists, contract is finalized into default, assumed settlement.
12. Maintenance Functions: Various maintenance functions to change the contract owners’ address, pause new contract creation, point to updated contract (if applicable), change contract fee rate, and maintenance override functions. Future implementations will include community decision making frameworks like a DAO to further prevent potential misuse of maintenance functionality on the protocol.
13. Reveal Information: In the case that TradLaw is selected for further appeal, the Arbitrator and Challengers information shall be exchanged to ensure prompt availability to serve notice. Either individual may select 3rd party individuals to represent them, if so, they must be identified immediately. If information is not correct or made available, automatic judgement will be made against this party.

**Future Development**

This protocol will require ongoing efforts and development to ensure that it is truly decentralized and trustworthy. The main deficiency currently is requiring a detailed security audit and stress testing of the contracts as current implementation likely does not account for many potential security vulnerabilities particularly given the complexities of interacting functionality. Additionally, there is a large variety of functions that could further drive utility and adoption of such a protocol these are but not limited too: Pooled contracts and multi-party contracts. Multiple Arbitrators providing consensus on decision making. Identity verification of arbitrators and qualifications to further prevent bad actors. Direct Bitcoin or Ethereum Settlement instead of BTCW and ETHW for layer-2 adoptions. Flexible contract settlement options allowing negotiated stances. DAO creation for decentralizing maintenance of protocol. Efficiency gains and streamlining to ensuring low gas fees on execution.

**Conclusion**

As we conclude this white paper, the Delaw protocol has the potential to significantly expand the potential uses cases for on-chain transactions and disrupt the current landscape of legal arbitration and contract resolution. Delaw paves the way for a new era of dispute resolution by creating a protocol that stands as a testament to the ingenuity of decentralized solutions. This protocol offers scalable, cost effective and accessible alternatives for a range of potential use cases from simplistic customer service disputes to complexities of insurance claims or real estate transactions.

To bring this vision forward, we ask for direct feedback from the Ethereum and Layer-2 communities to better incorporate multi-party contracts and more flexible settlement options. We aim to ensure proper market fit for this protocol and require the assistance of all members of our decentralized community to ensure our success. As an open-source project, we welcome all contributors who wish to assist us in this endeavor and we hope through use of the protocol that we will be able to compensate contributors through negligible maintenance fees applied to the network.

The heart of Delaw is a design that facilitates a trust-less environment where all participants can engage with confidence and be reasonably assured that the community will reciprocate in good faith (or be forced too).