**AvaxSportsBook**

Sports Betting Contract--Preliminary

Eric Falkenstein

V1.0 8/30/2023

**Abstract.** AvaxSportsBook is a decentralized blockchain contract for betting on weekend sporting events. Users can either bet or provide capital to the book. Cross-margining allows a small number of liquidity providers to support unlimited betting and diversify across events. Its exclusive oracle and liquidity providers equally share the standard sports betting vig.

Contents

[1 Introduction 2](#_Toc144113679)

[Simplicity from Restricted Focus 4](#_Toc144113680)

[Simplified Oracle 5](#_Toc144113681)

[Oracle Incentive Compatibility 5](#_Toc144113682)

[2 Contract Basics 7](#_Toc144113683)

[2.1 Outline 7](#_Toc144113684)

[2.2 Schedule and Start Times 9](#_Toc144113685)

[2.3 Odds 9](#_Toc144113686)

[2.4 Betting Capacity and Cross Margining 9](#_Toc144113687)

[2.5 Betting and Redeeming 10](#_Toc144113688)

[2.7 Liquidity Providers (LPs) 10](#_Toc144113689)

[2.8 Incidence Response Suggestions 11](#_Toc144113690)

[2.9 Avalanche 11](#_Toc144113691)

[3 Oracle Incentives 11](#_Toc144113692)

[3.1 Sending and Validating Oracle Data 12](#_Toc144113693)

[3.3 How Oracle Token Holders Claim Oracle's Revenue 13](#_Toc144113694)

[3.4 Tests 13](#_Toc144113695)

[4 Conclusion 13](#_Toc144113696)

[Appendix 15](#_Toc144113697)

[Odds Translation 15](#_Toc144113698)

[Redeeming a Bet 17](#_Toc144113699)

[Function Restrictions 17](#_Toc144113700)

[LP Rewards 18](#_Toc144113701)

[LP Eth to LP Shares to LP revenue 18](#_Toc144113702)

[Oracle avax Revenue 19](#_Toc144113703)

[Margin Adjustment for New Bet 20](#_Toc144113704)

[Gas for transactions 22](#_Toc144113705)

[Settlement Detail 23](#_Toc144113706)

[Odds Stability 24](#_Toc144113707)

# 1 Introduction

Sports betting is ideally suited for a completely on-chain smart contract. Consensus odds for straight bets on major events are statistically accurate and stable. The peculiar nature of sports odds—their stability and implicit bid-ask spread—avoids the adverse selection problem in standard asset swapping markets with stale quotes.[[1]](#footnote-1)A single human gabbing odds *in toto* from a large sports book will be sufficiently close to the optimal odds to prevent bettors from arbitraging the LPs. Bettors do not need *better* odds so much as easy access to ubiquitous conventional odds. If standard odds and the subsequent event outcomes are recorded on a blockchain smart contract, it is straightforward to apply escrow accounting logic where all bets are permissionless and secure. There is no need for price discovery, no economies of scale or scope in gathering correct data, no wisdom of the crowd.

For American football and mixed martial arts, the absense of intraweek matches makes it easier to assess the odds on the weekend. This creates a predictable repeated game for the oracle. The contract is completely self-contained, with all relevant data on the blockchain, which combined with its radical decentarlization makes it uncensorable. A contract targeting a limited but large portion of a multibillion-dollar market is small enough to manage and big enough to matter.

There are three types of *AvaxSportsBook* (hereafter, **ASB**) contract users: bettors, liquidity providers, and the oracle collective. Bettors can take either side of any regular bet offered, subject to a size constraint based on the amount of free liquidity provider (LP) capital. An oracle token holder submits a slate of matches, odds, and start times on Tuesday morning, and betting will be available by that evening until individual gametimes. The weekend’s event outcomes are sent to the contract the following Monday, which then loops through the 32 (max) events and settles that week’s betting. Redemptions by bettors are available immiately after settlement, while LPs cannot withdraw for 2 settlements after deposit.

Sportsbook odds are efficient, in that statistically the house makes a profit, but week-to-week the book can lose money due to small sample variation. The LP’s main risk, however, is the catostrophic risk via a hack, which would be a black swan event, something not detectible from its internal historical data. Thus, LPs have an incentive to become oracle token holders to align their incentives, as the most likely cheat would involve a conspiracy between the oracle and a bettor, defrauding the passive LPs. The token rewards allocation to the initial LPs encourages an LP/oracle overlap.

As LP capital is used to collateralize residual bet exposure, the ratio of the net to gross betting volume will be key in determining the return on LP capital. The amount of LP capital relative to the betting volume will equilibrate the market (e.g., if the return is too low, capital will leave, raising the expected LP returns).

Bets are automatically cross margined so that the capital required is minimized. For example, 10 AVAX collateralizes a single bet paying out 10 AVAX, and a contest where the winning payout is 510 AVAX if a team wins and 500 AVAX if its opponent wins.[[2]](#footnote-2) LP capital of X supports a book of infinite size when bets of size X are made sequentially. The required LP capital on any single event is a function of the maximum *net* payout, which is the payoff on a team *minus* the amount bet on its opponent. An adjustable parameter limits how much of this capital can be applied to any one event so that LPs are not subject to extreme concentrated risk (eg, if there are 100 AVAX, a parameter of 10 restricts the net LP liability for any one contest to 10 avax).

An exclusive ASB oracle provides and validates data sent to the betting contract. ASB fees are split evenly between the oracle and the LPs.[[3]](#footnote-3) Submissions are sent by a single token holder and the collective has at least 11 hours to vote before it can be sent to the betting contract. Token holders fee revenue is a function of how often they vote on data submissions. For example, if a token holder votes on 50% of the data proposals, she would receive half of her potential revenue. The forsaken oracle revenue is reallocated to the other token holders in the oracle contract.

Initially, the oracle will be relatively centralized in its initial handful of token holders. I chose three people, who do not know each other, to give 13.3% of the lifetime supply of tokens for two reasons. First, one of them could become incapacitated. Secondly, if one decides to defraud the contract, this improbable irrationality would countered by the other two who will have the ability and incentive to discipline him.

I created this dapp but I have no control or financial interest; I cannot disable or administer the contracts in any way.[[4]](#footnote-4) Contract users—token holders, LPs, bettors—are responsible for obeying their local laws and regulations. The fact that the only way for me to get this contract to work is to give it away is likely a major reason why no one else has created such a contract. The other 60% of the supply is set aside within the betting contract as rewards for initial LPs, so that each week LPs can get a pro-rata share of the token rewards available that week. If all the LPs claim rewards each week, the incentive program will last 20 weeks, otherwise it will last until all the tokens are distributed. There is no ability to mint more than the one billion initially minted.

Smaller governance token holders rarely participate in votes. This makes rational sense because there is a minimum fixed cost to evaluating a proposal, say 20 minutes of time.. Assuming 4 votes a week, the implicit hourly wage for this service would be under $1 for someone with less than $1000 worth of tokens. The voting incentive for small token holders would create an incentive for an automated voting protocol, which creates an attack surface for hackers.

Most token holders will be small and will not monitor and discipline data submissions for rational reasons. Such token holders create a deadweight loss for contract users. To mitigate this problem, the minimum deposit in the oracle contract is 40 million tokens, 4% of the total supply. Anyone with tokens on the oracle contract can vote on data submissions, but one needs 100 million to post data). Tokens must be in the oracle contract, and must vote, to receive fee income. This creates an incentive for small token holders to pool their tokens and designate a voter who would have an incentive to seriously monitor and discipline oracle data submissions.[[5]](#footnote-5) Like representative versus direct democracy, it’s the best solution to an perennial problem.

Sports betting is a competitive market, so the standard 4.5% vig reflects an equilibrium balancing the demands of bettors and bookies as opposed to monopoly power. By taking the standard vig as a given we remove naive schemes that in theory are less costly but in practice create failed markets like Augur. The hassle-free ability to bet or be the house should be sufficient to make it a dominant alternative for many sports bettors.

## Simplicity from Restricted Focus

### Weekend straight-up events

A focused set of weekly win/lose events makes incentivizing the oracle easier. In contrast, if we targeted a standard centralized sportsbook, it would cover diverse events on most days of the week, including exotic bets that are are not straightforward to validate. Realistically, only a subset of the oracle token holders would evaluate the large amount of data in your standard online sportsbook. This creates edge cases where a minority of token holders can take advantage of inattentive oracle token holders, a major attack surface. The weekly reporting also makes the oracle easier to monitor, in that the historical event logs refer to standard bets on weekend events.

Football, boxing and MMA will thus be the primary focus. The matches and odds are well-publicized early in the week. If there were a high profile events other than football and MMA can be accomodated on a case by case basis (for example, a soccer World Cup championship match).

### The contracts are all non-upgradeable

This removes any need for governance to vote on upgrades. Most importantly, it means there is no group of developers managing, promoting and proposing changes. Such developers would need to be paid, and generally this requires a corporate structure. Such corporations are attack surfaces for censors. removes risks from bugs often found in upgrades.

### Weekly settlement

Settlement can only occur the Monday following the next Friday.

### Maximum of one daily data submissions

The oracle processes at most one submission per day, which must be submitted during the 12th hour GMT. This makes it easier for the oracle to keep track of the data it must evaluate. Odds on major contests do not change considerably in the week up to a game, so a daily update should provide the LPs with sufficient protection against the adverse selection risk created by stale odds.

### No new data submissions on Saturday and Sunday.

This allows the oracle token collective to take time off without worry about a hack.

### At least 11 hours for data vetting.

As no healthy adult sleeps more than 10 hours a day, all token holders will be able to vote before the data submission is processed.

### Maximum 32 events per weekend.

Settlement loops through the events, and 32 is big enough to capture most weekend events.

### No ex post disputes requiring adjudication.

A protocol for disputing data validated by the oracle generates considerable delay. The oracle incentives are based on the present value of the oracle token, which should be sufficient. Redundant mechanisms lessen the incentive for the oracle token holders to monitor and discipline data submittors.

### No stablecoins

By using native AVAX for all bets we eliminate unnecessary costly swapping into stablecoins. As stablecoins are generally centralized, we eliminate an attack surface.

### LPs and token holders cannot withdraw for at least one week.

Flash transactions supposedly enable efficient arbitrage, but invariably introduce attack surfaces.

### All relevant data are on the Avalanche C-chain.

There is no way to censor a contract that is completely on-chain.

### Extreme odds are not allowed.

Matches with extreme payouts are attractive for hackers, as the generate the most revenue for the smallest amount of capital. It is difficult to empirically assess these odds, and they attract hackers looking for large returns on small investments. Initial decimal odds on favorites greater than 7:1 are not accepted. Initial decimal odds for favorites must be greater than 1.150, which implies an 88% probability of a win. Events with odds near this will be excluded (this covers about 5% of NFL games historically).

## Simplified Oracle

Hardest part is gettingthe slate of matches.

Automate as much as possible, but this only goes so far.

creating the slate:

grab matches, odds, and start times

macro to create input files sent to the oracle contract. Should be automated. Automatically translates times into UTC, the pair odds on a match into the unique single odds used by this contract.

python files to automate the submission of this data so one does not need to use a GUI or Remix. Put into a cronjob or scheduled task.

# Oracle Incentive Compatibility

Incentive compatibility is vital to low-cost enforcement of contracts, and historically this mechanism centered on reputation, not contract law administered by the state.[[6]](#footnote-6) The blockchain's transparency, immutability, and pseudonymity make reputation much easier to monitor. When agents have incentives aligned with their counterparties, we minimize indirect costs (delay and spread), making it more attractive for bettors. If the bettors and LPs want to cheat, they need to collude with the oracle; if the oracle does not cheat, bettors and LPs cannot cheat. The focus is to ensure honest and timely reporting of odds and results is always the oracle collective's value-maximizing act.

In the prisoner’s dilemma game, the Nash equilibrium strategy is for both players to play the noncooperative strategy. In the movie *A Beautiful Mind* this insight supposedly turned economics on its head because in large markets the standard result is that competitive market outcomes are socially optimal.[[7]](#footnote-7) In reality, the Nash equilibrium did not invalidate Adam Smith’s invisible hand, it just highlighted the importance of repeated interactions for lowering transaction costs. In the *iterated* prisoner's dilemma, the equilibrium strategy is to cooperate and so acheives the socially optimum strategy. A properly incented game ensures that for all players at all times, cooperating has a higher present value than cheating. This is selfishly motivated reciprocal altruism—I play nice because I expect you to play nice—enforced by the threat of punishment.

Creating a game where honesty is an oracle's best strategy is straightforward; the keys are simplicity and a repeated game, which leads to easy monitoring and credible punishment anticipation for a fraudulent oracle data submission. Additional parties, tokens, or scope increases cost, complexity, and delay. Defecting/defaulting implies the loss of the benefit of future interactions, the present value of which acts like over-collateralization on a loan. The potential cheat gain is less than the present value of future revenue foregone even if the oracle were centralized. Incentive compatibility is critical to low-cost enforcement of contracts, and historically this centered on reputation, not contract law administered by the state.

Consider the following cost-benefit analysis for ASB's oracle. A conservative equity price/earnings (P/E) ratio is 10. Assume a betting contract has 100 in net exposure, which we will conservatively assume is the book’s average gross betting exposure (no offsetting bets that generate reward but no risk to the LPs). As the oracle's fee is about half of the vig, this would average about 2.5 AVAX in weekly revenue. Given 50 settlement events over the year, this annualizes to 125 AVAX. Given a 10 price/earnings ratio, this values the oracle collective at 1,250 AVAX. The maximum potential cheating revenue in this example is 100 AVAX, so the LPs have net exposure to the wrong side of every bet made by the cheating oracle’s sock puppet bettor.

Such a scam would be conspicuous in the readable event logs, and no rational person would use this contract again, making the value of the oracle token zero. Unlike oracles that service many contracts, there is no plausible deniability by the oracle. That is, the oracle is ‘all in’ on this betting contract. There is no reason for allowing incorrect data to get voted to the betting contract outside of a conscious intent or radical incompetence, especially given the low verification costs generated by the restrictions on timing and event coverage, and how all oracle voters will have significant token stakes (at least 4% of outstanding tokens). A voting majority's oracle token has a present value of 625 AVAX in the above example, significantly more than the 100 AVAX in a cheat.[[8]](#footnote-8) Honest reporting is the dominant strategy in the improbable worst-case scenario described above.

The oracle voters have, literally, all day to evaluate a data submission that can be evaluated in a couple of minutes. A majority 'no' vote among token holders cast penalizes the proposer of the data.[[9]](#footnote-9) The limited focus of events, timing of oracle submissions remove any plausible deniability for the oracle cheat action in any single event each week. I provide a spreadsheet where a noob can input the weekend’s matches and generate the relevant data into a text file, which a python program then sends to the contract. As there is only one submission allowed per day, and a single oracle token holder sends the data, the average oracle token holder merely evaluates the day’s submission for obvious fraud, and then votes with a click on the website or python programs in the GitHub. The website generates event log data in readable form, so one does not need specialized knowledge of hash functions to evaluate the oracle’s behavior.

While it is simple enough to incent the oracle properly, this only protects the contract against insiders. In contrast, decentralization defends this contract against outsiders. Powerful institutions have always used centralized power to prevent competition, often using disingenuous rationales emphasizing safety. Such an attack needs a choke point, prevented if a collective of pseudonymous accounts worldwide administers the oracle. If the oracle is profitable, an infinite number of people will be able and incentivized to replace token holders captured by outside attackers.

# 2 Contract Basics

## 2.1 Outline

Event data, including odds, are sent to the contract on Tuesday, allowing people to bet on weekend events from Wednesday to game time (which is loaded in UTC with the odds). The subsequent Monday, the oracle sends the results to the betting contract, settling that week's bets, and the contract then repeats the process.

Bettors and LPs need only interact only with the betting contract, while the token holders only interact with the oracle contract. All transactions with the betting contract are denominated in AVAX, including providing liquidity and betting.

The LP capital backstops residual imbalances in the book. The LP's total capital is available for all contests that week, and there is a limiting mechanism on how much AVAX can be allocated to any single contest. This diversifies the LPs, reducing the chance that a single contest outcome could extinguish LP capital. The betting contract contains all the methods for bettors and LPs: betting and redemption for bettors, investing, and withdrawal for LPs.

A singular oracle token holder, with at least 10% of the oracle tokens outstanding, proposes the relevant data: the upcoming schedule (who plays whom, when), odds, results, and some technical parameters (e.g., paused matches, the diversification parameter). Each submission is then subject to an evaluation period subject to a majority vote: send or reject. A successful data submission is sent to the betting contract after a voting period of 11 hours, giving the oracle collective sufficient time to veto a fraudulent or incompetent submission.

odds, results, etc

avax fees

Oracle Contract

Equity Token Holders

Betting Contract

avax

avax

data, votes

avax

tokens

LPs

Bettors

**Weekly Schedule**

Standard Oracle Actions

|  |  |  |
| --- | --- | --- |
| **Day** | **GMT Hour** | **Oracle Action** |
| Tuesday | 12 (ie, 12:00-13:00) | post Schedule, start times, odds |
| Wednesday | AM (ie, <12:00) | process |
| Wednesday | 12 | post odds update |
| Thursday | AM | process |
| Thursday | 12 | post odds update |
| Friday | AM | process |
| Friday | 12 | post odds update |
| Saturday | AM | process |
| Saturday | 12 | nothing |
| Sunday | AM | nothing |
| Sunday | 12 | nothing |
| Monday | AM | nothing |
| Monday | 12 | post results |
| Tuesday | AM | process |

The data submitter automatically votes for his submission, so if no one votes, it will succeed, as it is a simple majority vote that determines success or fail. Posting data can only occur in the 60-minute window where the GMT hour is between 12:00:00 and 13:00:00 GMT, which is 8 PM New York time in the summer and 7 PM in the winter. Voting takes place between the noon posting and processing the post, which can take place at any time, by anyone, at a GMT hour less than 12, which starts right after midnight GMT.

If a settlement or initial post are rejected, then a settlement or initial post, must be posted again. This effectively delays the contract by a day. Thus, the days listed above can be changed if a post is rejected. For odds updates, these are optional and can be omitted. Odds updates cannot occur Saturday or Sunday, and settlement can only be posted the following Monday, so the oracle token holders have no duties until Monday’s settlement.

Each week, aka ‘epoch,’ the MMA, boxing and US football games that weekend are sent to the oracle contract. The data are sent to the betting contract if most oracle token holders vote yes. When the betting contract has data for the next weekend, bettors can bet up to the time of the various games that weekend, as each game is given a starting time. Odds can be updated, but only once a day, and only in the same 60-minute window each day.

After the weekend, the outcomes are sent to the betting contract, which settles all the week's bets. Once the settlement is sent to the betting contract bettors can redeem their bets. The contract has no ability to seize neglected funds, so as long as the blockchain exists, users will be able to safely let unredeemed money sit in the contract. LPs cannot withdraw or deposit from Friday 7 PM ET through settlement Monday evenining, as otherwise, they could game the contract by anticipating unusual losses or winnings.

## 2.2 Schedule and Start Times

Each betting period will contain up to 32 events and target a weekend (e.g., Friday night through Sunday night). Each contest is slotted into an array that can be unambiguously linked to its outcome via event logs that expose what events odds were on the contract. The schedule array contains a string with the sport (NFL, MMA, etc.), the two opponents, and the starting time. The initial favorite will be listed first and the underdog second, though the odds can change over the week while the ordering of the contestants cannot.[[10]](#footnote-10)

The start time is important because if it is wrong, bettors will be able to bet on games that have either started or completed. Websites with event start times are tricky because sometimes these are listed as ET (ie, New York City time), but sometimes they are automatically converted into one’s regional time zone. It is essential to know what time zone is represented in a time of day, as the number sent to the contract uses the universal GMT UTC.

Start times are more ambiguous for UFC events, as often they are held in Dubai or Singapore and so occur at unusual times in the Americas. For example, last Saturday the main event for UFC was held around 9:30 AM central time.

## 2.3 Odds

The ASB.co frontend presents all odds in terms of payout to the bettor, the 'all-in' odds that include the fee to the oracle. Thus, when a bettor sees 2.000 decimal odds in the ASB GUI they can be sure they will receive 2.0 AVAX if they bet 1.0 AVAX, etc. For such odds, the contract itself would have 1 + 1.00/0.95 or 2.053, which is the gross odds before the 5% oracle fee is applied to the payout. The oracle may update odds over the week, but the odds posted on the contract at the time of the bet are applied.

## 2.4 Betting Capacity and Cross Margining

The contract’s logic makes sure all bets are fully collateralized. Unlike in futures markets, the margin is not derived from a probabilistic risk, such as an instant 20% price movement. Such a rule exposes a contract to insolvency, as there is a possibility of price movements greater than 20%, which becomes increasingly probable over time. In betting on binary outcomes, the worst-case scenario for the house is assumed, which is like assuming each LP net position loses. The contract will always be fully collateralized on all bets, as this is enforced at the time of each bet. There is no insolvency risk.

As bettors take the opposite side of a contest, it is a waste of capital to require the LPs to collateralize both sides independently. The solution involves netting exposure.

**Cross margining example:** Assume two teams are given even odds so that for either team, a 1 AVAX bet pays the winner 2 AVAX. If there is 10 AVAX on team A, and 10 AVAX on its opponent, team B, it would be a 'flat' book in that the LPs have no exposure to this game; payoffs are funded by betting counterparties, not the LPs. A new bet that pushes the book to have a net exposure would necessitate LP funds as collateral, so a bet of 2 AVAX on team A would move 2 AVAX from the LP's free margin to the LP's locked margin. Given a total of 12 on team A, and 10 on team B, a bet of 2 AVAX on team B would move 2 AVAX *out* of the required margin to the free margin because the resulting book would be flat again.

Adjusting the net required LP margin involves 'linear programming' where the LP's net game exposure is the maximum liability of either team winning. The margin adjustment is applied at the time of a bet, so there must be sufficient free LP collateral to accommodate a bet adding to LP exposure.

A contract parameter prevents an overconcentration of LP capital on one event. For example, 123 AVAX in total LP capital and a concentration parameter of 10 implies a maximum of 12.3 AVAX LP exposure for any event. Thus, if the current LP liability for team 0 winning was 10.0 AVAX, it could only accommodate an additional payout of up to 2.3 AVAX on team 0. In contrast, a bet on team 1 could accommodate a bet payoff of 22.3 AVAX. This concentration parameter can be adjusted over time by large oracle holders.

## 2.5 Betting and Redeeming

All ties and 'no contest' games give bettors their initial bet back. Winners receive their bet amount plus the payoff implied by their bet odds. When the weekend's results are sent to the betting contract, all bets are settled, and the oracle payment is sent to the oracle contract. Settlement creates a mapping from a unique epoch-match-team, which determines which bets credit a user’s balance when looping through a bettors bets. Settlements should happen Monday evening.

Bettors redeem all of their outstanding bets in batch. The redeem function loops through up to 16 bets in a user’s account, and sends the winnings to the bettor’s account. Redemption can only be processed if there are no active bets in the account, so bettors must wait until the next settlement to redeem if they have any active bets. If an account has 16 unredeemed bets, it must redeem them before it can place another bet.

All bets are fully collateralized, and so are the accruals for the oracle. so if one has access to the account used for sending the bet or LP investment, a user's funds are safe in the contract. Unclaimed bets or tokens can reside in the contract and retain their value as long as the AVAX C-Chain exists, as there is no mechanism to sweep neglected funds to LPs or oracle token holders.

## 2.7 Liquidity Providers (LPs)

To become an LP, one sends AVAX to the betting contract, which then credits the LP with shares representing their pro-rata ownership of the LP pool. For example, if there is 10 AVAX in the LP book, and 10 shares, adding 1 AVAX would give a new LP a 1/11th share of the new pool of 11 AVAX, keeping the AVAX/share the same. This LP claim exists only within the betting contract and is tied to the initial LP AVAX account address. It is not transferable to other AVAX addresses, and so is not represented by a token. The size of the LP capital should adjust to the volume and degree of cross-margined trading, which will determine the expected pnl. The relation of an LP's share value will be equilibrated by its net capital, which makes creating an LP share token problematic.

LPs can only withdraw during the inactive period if margin is available. A bet backed by LP margin locks this margin until settlement, or bettors take the other side, freeing the LP capital. Since there is at least a 24-hour window each week after settlement (~Monday 8 PM ET) before new bets are offered (~ Tuesday 8PM ET), LPs are sure to be able to withdraw at least once each week. More practically, there will be free margin available for marginal LPs to withdraw over much of every week, as bettors will probably not max out the bookie's free margin in the first days of the week.

LPs must also have their AVAX in the contract for at least two settlements. If LPs could withdraw after only one settlement, people could add large amounts of capital just before the weekend when the pool has little net betting exposure, and then take their money out right after settlement. For example, if all bets were fully collateralized by offsetting bettors one week, the LPs would receive a certain profit given the vig built into the odds. Outsiders could provide superfluous liquidity just before the weekend and withdraw immediately after settlement, generating a certain profit. This would dilute the profits of LPs providing 'real' liquidity. Therefore, an LP must expose herself to at least one complete betting period before withdrawing (i.e., two settlements).

## 2.8 Incidence Response Suggestions

There is no outside adjudicator to rectify problems, as this would delay payments and complicate the contract (how to incent the adjudicator?). All problems must be solved on-chain within these contracts. As I do not and will not control the contract in any way, this is advice rather than an official protocol or something automatically enforced within the contract code.

Off-chain odds can change quickly and significantly, exposing the LPs to arbitrage. In that case, oracle token holders can immediately pause up to two bets. This action does not require the usual 12-hour vetting period to allow oracle token voting. It does not expose LPs or bettors to more risk, as it just prevents new bets on those matches, which will be allowed again if new odds are posted.

A more extreme solution to bad data on the betting contract would be to adjust the concentration factor. Remember that the maximum exposure on any one match is the total LP capital divided by the concentration factor. If one sets this to a large number, this number is effectively zero, and so would prevent *any* bets increasing LP exposure. Again, this can be done by large token holders, it simply prevents more risk, and can be rescinded when new odds are pushed to the betting contract.

If a hacker could sneak in bad odds that enabled a cheat, the oracle collective could nullify this action by posting a result of a 'tie' regardless of the outcome. This allows the LPs and bettors to get their money back as if nothing happened, and the ‘incorrect’ but fair tie result should be explainable by the event logs showing the earlier hack. This would be an extreme scenario, like a fork in a blockchain, but it is helpful to anticipate.

# 2.9 Avalanche

Snowball is Avalanche’s proprietary Proof-of-Stake (PoS) consensus mechanism. It consists of a combination of Avalanche’s Directed Acyclic Graph (DAG) architecture, the system’s Slush consensus (single-decree consensus), and Snowflake (Byzantine Fault Tolerance-based) mechanisms. Its efficiency makes it as efficient as an Ethereum Layer 2 blockchain like Optimism. Unlike Ethereum’s Layer 2s, however, it is already decentralized. In May 2023 Ethereum saw transaction costs spike 6-fold. While some L2s and this spilled into their L2s. To the extent it did not, that just highlights how their blockchains are subsidizing users, which is possible because their various tokens have sufficient value. To be sustainable, these L2s will have to charge more, and it is uncertain whether or not that will work. Avalanche’s gas price doubled on June 10 2023, but otherwise have remained stable over the past year, with gas prices rarely moving more than 30%.

Avalanche uses the same Ethereum Virtual Machine as Ethereum, so it took no extra work to port the contract onto this non-Ethereum blockchain. It also has the same address structure as Ethereum, so users can use their MetaMask wallets to store and transact with Avax. Their Core Wallet, however, makes bridging assets from Ethereum or Bitcoin safer and easier than Ethereum’s L2s.

Avalanche’s blockchain is relatively fast, 2 seconds vs. 10 seconds for Ethereum. This makes it more difficult to query event logs, as there are so many blocks to process, but this is a solvable problem.

# 3 Oracle Incentives

## 3.1 Sending and Validating Oracle Data

The oracle token holders get paid for performing a specific purpose, enforcing honesty. Other than the singular data submitter, evaluating the data and clicking a button to vote on it imposes an insignificant cost on the oracle. In an ideal world, one should not have to pay someone to *not* be evil, but that is not the world we live in; the line separating good and evil passes through every human heart, and crypto has shown that anything pseudonymous based on trust will be exploiting by amoral hackers to the fullest extent. Thus we make sure honesty is the oracle’s profit maximizing action at all times. They can then spend that money on whatever they like, wholesome or debased.

A token holder needs at least 10% of the outstanding tokens to submit data to the oracle contract. A power law distribution will always accrue in ownership, and the top owners should find it in their best interest to lead the oracle administration. While submitting data takes some effort, the cost is relatively low given the ease at which relevant data is available and the limited scope and frequency of data submissions. Larger token holders should be sufficiently motivated to send data to the contract promptly.

Tokens must be deposited within the Oracle contract to submit or vote on submissions. This prevents double-voting and forces the token holders to attend to the contract they should be monitoring. The tokens are meant for governance, not speculation, and generate dividends directly proportional to the bet volume. Token holders cannot vote more than once on any data submission, which requires that token holders cannot withdraw tokens while a vote is active.

all times 12:00-13:00 GMT Data vote Outcome

Tuesday Initial slate pass

Wednesday\* odds update pass

Thursday\* odds update pass

Friday\* odds update pass

Monday settlement pass

Tues Initial slate pass

etc.

\*optional

all times 12:00-13:00 GMT Data vote Outcome

Tuesday Initial slate pass

Wednesday odds update pass

Thursday odds update fail

Friday odds update fail

Monday settlement fail

Tues settlement pass

Wednesday odds update pass

Initial data proposals must be sent between 12:00 and 13:00 GMT (8-9 AM summer New York), and votes are not counted until midnight. This gives oracle token holders sufficient time each day to see each data submission. Anyone can execute the function that processes the vote. Upon submission, the proposal can be submitted to a vote count at any time after 18:00 GMT. A yes vote sends the data to the betting contract, while a no vote burns a fraction of the proposer's bond and resets the state for the next data proposal.[[11]](#footnote-11) For rejected initial and settlement data submissions, these must be resubmitted; for a rejected odds update, they can be resubmitted or not.

Token holders submitting failed data submissions are charged a fee equal to 0.25% of the immutable total supply. Rejections should be rare, but they should happen occasional due to unintentional mistakes. The penalty is meant to be painful but tolerable, as three initial oracle token holders control 13.3% of the token supply, and the minimum token holding needed for submitting data is 10.0%. Those rejecting data do not get a bonus, as we do not want to incent users to reject data submissions to acquire more tokens. In the extreme case where the large token holders have close to the minimum data submission amount, they can adjust the minimum data submission parameter to accommodate them with their fewer tokens.

## 3.3 How Oracle Token Holders Claim Oracle's Revenue

Each week the oracle receives 5% of the bettor winnings as a fee for their service. This is consistent with the Oracle receiving 2.5% of bet amounts and the LPs receiving the other 2.5%. While the oracle could receive nothing if all bettors lose, the oracle can never lose money. Token holders must vote on data submissions to get their revenue, which requires they keep their tokens in the oracle contract. For example, if their tokens were in the contract for three settlements, and they voted twice, they would receive one-third of their payment. The other two-thirds would be added back to the pool to go to the other token holders, incenting the token holders to actively monitor the the data submitted.

The primary way the oracle acrues fee revenue is at settlement. The epoch incements each settlement, which is why this number is recorded in a token depositor’s account. The other way the oracle acrues revenue is when negligent token depositors claim revenue. The unearned AVAX is then immediately reallocated to the other token holders by treating it like a settlement dividend

## 3.4 Tests

On GitHub, there are several test scripts documenting the integrity of the contract methods. These JavaScript files correspond to hypothetical transaction sequences documented in ASBtests.xls, and are described in ASBtests.docx.

# 4 Conclusion

Most sports betting sites touting their crypto functionality are conventional ones accepting crypto. A truly blockchain-based betting dapp upholds Satoshi's vision of *pseudonymity*, *confiscation-proofness*, and *permissionless access*, which requires it to have no off-chain presence. I hope that a focused dapp with good incentives can provide an example of what works on the blockchain. The purpose of the contract is to facilitate betting, not create a new token for speculation. Unlike most dapps, where tokens have a vague governance role and hypothetical revenue, ASB’s token holders have a straightforward but essential job that generates instant revenue.

One should expect players to always act in their selfish best interest. A sustainable contract creates a repeated game where honesty is always the dominant strategy for every player. Simplicity is crucial in generating the cooperative equilibrium because the state space grows exponentially in the number of players and actions they can take. An incentive-compatible contract avoids costly and redundant adjudication procedures, allowing bettors to cash out in timely fashion. The trust one puts into the ASB Oracle is fundamentally the same as why investors trust miners: the rational self-interested assessment that honesty dominates dishonesty for a hypothetical individual.

Blockchain betting contracts are the perfect application of straightforward rules to a common use case. ASB presents a quick and efficient way to get asset exposure without the many hassles in standard contracts. Sports betting is ubiquitous, but it should be easier. This contract provides a simple way to do that.

# Appendix

## Odds Translation

To convert moneyline odds into Decimal odds, we have the following.

For positive moneyline odds: (Moneyline odds/100) + 1 = Decimal odds

For negative moneyline odds: (100/Moneyline odds) + 1 = Decimal odds

To translate decimal odds into moneyline odds that are prominent on NFL betting sites, we have the following adjustment mechanism:

If decimal odds are greater than 2.0: 100 × (decimal odds – 1) = Moneyline odds

If decimal odds are less than 2.0: -100/(decimal odds -1) = Moneyline odds

To translate moneyline odds to fractional odds:

For positive moneyline odds: Moneyline odds/100 = Fractional odds

For negative moneyline odds: -100/Moneyline odds = Fractional odds

To convert decimal odds into winning probability.

prob(win) = 1 / Decimal odds

Decimal odds = 1 / prob(win)

The most common odds offered for the NFL are presented in moneyline form as ‑110 for both teams. A flat book on such a wager would receive 220 and payout 210. In this way, the 'house' makes money used to pay for various costs and a profit from the house. The implicit profit ('vig') in this case would be 4.55%, 10/220. The general formula for estimating the vig is given by the following formula, where *p* and *q* are decimal payouts (e.g., 1.909 for a standard even money bet) for opposing teams.



The spreadsheet 'ASB.xlsx' presents a page where people can see how these transformations are applied. Those interested in sending odds to the contract will find it a helpful template.

**Odds in the contract**

Odds are available on many betting websites, and arbitrage limits how far these odds can differ. On average, a team's implied probability of winning will change by only 2% over the week, rarely over 5%. All odd postings and updates are recorded in event logs, observable in online queries at sportAVAX.co.

A contest will have a single odds number posted for a contest. These odds are supplied only for the initial favorite using a truncation of preliminary decimal odds. For example, 1.909 would be stored as 909, 2.50 as 1500, etc. This number, however, is just relevant to the team in slot 0, the initial favorite. Further, it needs to be adjusted to reflect the oracle fee that would be assessed. Thus, the betting odds for a favorite where the match odds were 955 would be 910, via

Bettor Odds (favorite) = (contractMatchOdds \* 0.95)/1000 + 1

= 955\*0.95/1000+1=1.907

The odds for its opponent are generated within the contract by the following formula:



Then to account for the oracle take, the all-in odds for team 1 would be

Bettor Odds (underdog) = (underdogOdds \* 0.95)/1000 + 1

With this method, we can ensure that the set of odds for a contest generates a positive vig, removing a potential attack vector.[[12]](#footnote-12) This formula generates a vig of 2.5% for the LPs via parameter 45 in the above equation, and the 5% take of winnings generates an approximate 2.5% vig for the oracle.

The standard odds presented by most casinos embody the standard vig of 4.5%. For example, -110 on an even-money bet, which is 1.909 in decimal odds. This contractapplies the vig to each contest via an algorithm that works well, though it is simply a hack as opposed to something developed via first principles. The approach is to put the LP’s take into the ‘spread’ between the odds offered for the favorite and the underdog, but leave the Oracle’s take out of the spread. The Oracle fee is then taken out of the winnings.

For example, if the NYGiants are at 1.616 decimal odds. 1.909 on both teams in a match is consistent with a 4.5% vig. This is loaded as 957. If one bets 100 avax with a 1.909 odds, the winnings are 90.9 avax; 5% of that is 4.5 avax, which is sent to the oracle contract at settlement. Anticipating this, we present the odds in the GUI to 909.

Initial odds presented must be between 1.999 and 1.125. The cap at 2.000 reflect the fact the initial odds apply to the initial favorite, while the 1.125 minimum removes events where the initial odds are greater than 8:1, lopsided contests. Eliminating high payout contests mitigates risk, as such events would invite hacker focus. Further, these events are rare, and so are more difficult to validate historically.

The excel spreadsheet is provided that generates the data in the necessary format. The basic algorithm is this

1. Take an initial set of odds
   * Home team: +135
   * Away team: -150
2. rearrange so that the favorite team is first
   1. team[0]: -150, team[1]: +135
3. Translate into win probability
   1. team[0]: 59.8%, team[1]: 42.4%
4. calculate probability spread
   1. spread = 0.598 – 0.424 = 0.174
5. Calculate new favorite (team[0]) prob(win)
   1. prob(team[0] win) = 51.1% + spread/2
   2. prob(team[0] win) = 51.1% + 8.7% = 59.8%
6. Translate prob(team[0] win) into decimal odds.
   1. decOdds(team[0]) = 1 / 0.598 = 1.6716
7. Translate decimal into the payoff of the bet. This number will represent the match odds in the contract.
   1. payoff(decOdds) = 1000 \* (decOdds – 1)
   2. payoff(1.6716) = 671
8. Translate into netDecimalOdds(team[0]) presented to bettor
   1. 0.95\*671/1000 + 1 = 1.637
   2. The oracle fee reduces the actual decimal odds returns presented within the contract
9. Translate into team[1] payoff
   1. payoff(team[1])= 1e6 / (671 + 45) -45 = 1359
   2. netDecimalOdds(team[1]) = 0.95\*1359/1000 + 1 = 2.291

In this example, the vig is 4.5: 1.637 \* 2.291 / (2.291 + 1.637). The above algorithm generates a vig near 4.5% across the range of odds covered in this contract. Contests with win probabilities for either team greater than 90% are not permitted for several reason: would be most attractive for a hack, there are not many bets with these odds so it is difficult to assess their efficiency, and I did

The website sportAVAX.co displays the decimal odds users receive if they win. For example, a user seeing odds of 1.900 will receive back 1.900 times their bet amount.

## Redeeming a Bet

Bets are stored in a 16-element array, and after 16 bets, no further bets can be made until they are redeemed. Redemptions can only occur when a bettor has no active bets, so a bettor should redeem his bets after settlement if he anticipates a problem. All bets in the array are settled for the bettor. Each bet is represented by the unique combination of epoch, match, and pick. At settlement, a bets hash refers to a struct containing this information, and a mapping generated at settlement allows redemption. SportAVAX.co provides this data by reading the user's MetaMask address, allowing users to redeem bets by clicking a single button (it is one transaction). However, anyone can log onto the blockchain using the account used to make a bet and submit the bet hash.

## Function Restrictions

The most basic restriction is that the three data submissions can only be applied during the 12th hour on any day. No other submissions can be made until this is processed, and token holders have 11 hours to evaluate the data. After midnight, anyone can then process the vote, and if it passes the data are sent to the betting contract, while if it fails the process starts over. Thus, if a vote fails, this will push back the contract’s starting bets or settlement by at leat one day.

reviewStatus is a variable in the oracle contract that prevents the contract from updating in the wrong order. For example, after sending the initial set of matches, start times, and odds, one can only add an update to the odds, or the outcomes of those matches; one should not send another batch of matches, as this would make any existing bets point to different matches. Further, one cannot send new odds or results until the initial set of matches was processed via a successful vote. Additionally, token holders cannot withdraw while a data submission is under review, as this makes it easy to ensure each token holder votes only once on each data submission.

reviewStatus (RS) variable as control mechanism for constraining the data flow

required output vote succeeds vote fails

initPost 0 10 2 0

updatePost 2 20 2 2

settlePost 2 30 0 2

voteProcess 30 0

withdrawTokens <10

When the initial data are sent, the first start time is used to find the nearest Friday 7 PM ET, and this time prevents LPs from withdrawing until settlement is processed Monday night. This first start time also prevents any odds updates on the weekends, as odds should be sufficiently stable over the next 48 hours, and it allows the oracle token holders to rest. Finally, the outcome cannot be sent until Monday, in the 12th hour GMT. This prevents a surprise data submission that could sneak by the oracle.

Odds sent must be greater than 1.125 and less than 2.00. These are for the initial favorite, so the cap at 2.00 enforces this. Restricting matches where an initial odds are less than 1.125 for the favorite removes long-shot bets that are most subject to fraud, and are sufficiently infrequent to be irrelevant.

## LP Rewards

## LP Revenue

LPs own a pro-rata portion of the contract's revenue based on their percentage of LP capital before that week's events. Statistically, the LP capital will grow each settlement due to the vig; this is how LPs make money. As the relevant LP credit/debit occurs at settlement, the LP's AVAX/share value is fixed each week when users can withdraw or invest.

An initial investment generates the following shares:

LPshares = AVAX invested × TotalLpShares / TotalLpAvax

For example, assume the contract has 123 AVAX owned by its LPs, who have 100 shares. This AVAX may be sitting free or locked up as collateral for upcoming contests. This implies each LP share is worth 1.23 AVAX.

LP AVAX LP TotalShares avax/Share

123 100 1.23

Suppose Alice wishes to invest 10 AVAX into this pool. The above formula implies she would receive 8.13 shares (10/1.23). This would change the pool's balance sheet to

LP AVAX LP TotalShares avax/Share

133 108.13 1.23

Note the ratio of AVAX/share is the same after Alice's investment, so existing shareholders do not lose or gain money via Alice's new investment.

If we assume the LP collective gained 2 AVAX that week, the new balance sheet after a settlement will look like this:

LP AVAX LP TotalShares avax/Share

135 108.13 1.25

The increase from 133 to 135 reflects a 1.5% profit from that epoch's games. If Alice then sold her shares, she would receive AVAX using a transformation of the above formula:

avax Withdrawal = TotalLpAvax × SharesSold / TotalLpShares

Selling 8.13 shares would generate 10.15 AVAX, a 1.5% return on their investment, identical to how much the AVAX LP pool rose over that period.

In this way, any LP investment or withdrawal reflects the percent change in the size of the LP pool over the investment period.

One can see this in the contract tests

## Oracle Revenue

Oracle token holders must deposit their tokens in the oracle contract to vote, and then are expected to vote at least twice a week. The two fundamental data submissions are the initial set of contests, odds, and start times, and then the results of those contests. When a weekly settlement transaction is executed, the oracle's 5% fee is applied to the winnings and sent to the oracle contract. The '*feePool'* state variable reflects the lifetime amount of AVAX per token paid to the oracle contract.



When an oracle token holder deposits into the contract, their account notes the current value of *feePool*. When that oracle token holder withdraws or adds to their account, the token holder is sent their entire accrued AVAX using the formula



Having tokens in the oracle is a necessary but insufficient condition for being paid. The contract then takes the total number of tokens







This account's *OraclePoughback* is then applied to the Oracle *feePool* as if it were revenue from the epoch’s settlement. This gives an incentive for token holders to wait as long as possible to retreive fees, as unclaimed fees are secure: there is no scenario where the token holders can lose accrued revenue, either due to risk or an oracle hack. is then updated to the currentFeePool, so another immediate add or withdrawal of a token by the same token holder would have *CurrentFeePool* – *UserOldFeePool*=0, and receive nothing.

With this method token holders can be sure the contract is in balance, where accounts payable are equal to AVAX in the contract at all times.

## Margin Adjustment for New Bet

There are three types of margin tracked by the contract, all held in the array variable' margin.'

**LP Capital**

This is AVAX owned by the LPs, both free and locked up as collateral.

**LP Locked Capital**

This is AVAX owned by the LPs that are unavailable for bookie withdrawal. It represents the gross worst-case scenario loss for the LPs.

**Bettor Capital**.

These are bettor funds applied to outstanding, taken, bets. Bettor positions are not cross margined.

LPstruct. Mapping between an address and its share amount and time of investment. The total number of LPStruct.shares will equal sharesOutstanding, reflecting the percent ownership of an address of the bookie capital.

Free LP capital in margin [0] is available for new bets that increase the contract's net liability. New bets that increase the contract's net position will transfer AVAX out of margin [0] into margin[1]. Bets that decrease the bookie's net position will move AVAX from Margin [1] to margin [0]

For a team with decimal odds of 1.957. the total payoff for a win can be separated into two components: 1 + 0.957, the latter term representing the bettors net profit, and the former term representing the bettor's initial bet. The amount 1.000\*betAmount is available from the bettor funds, while 0.957\*betAmount must come from the LPs or bettors taking the other side. Odds are stored such that



LP Required Margin is the sum of the maximum liability for all the events in an epoch. Each event is independent, so the book is correctly margined by correctly margining all the individual bets. Thus we need merely describe how margining occurs for a single event, knowing these are then summed for determining the overall Required Margin.

The total amount owed if team 0 wins equal the sum of the bet amount and its payoff for all the bets taken on team 0. Let us define two types of capital used to pay bettors, the payout or profit, with must come from someone other than the bettor, and the bettor's initial bet amount, which is returned with his profit:



**betSum0** is the total amount bet on team 0, summing over all the bets on 0. Bettor funds are available for payout but not part of the LP's Required margin (which is in Margin [0] and Margin [1]). **paySum0** is the sum of the bettor's profit if team 0 wins, which requires AVAX from the LPs or bettors taking the other side. As the betSum of team 0's opponent, team 1, is available for paySum0, the trick is monitoring the ability to cover the LP's liability given the amount bet on its opponent. This generates the following maximum liability for the LP (*aka* required capital) for a contest in that it is the maximum liability to either team in a contest:



We add the zero term because the house will have only non-negative liability on every contest. For a new bet long on team 0 playing, the new bet and payout are added to the above max() equation and compared to the extant maximum liability. The difference is the change in the LP's required margin (margin[1]), which is offset by a change in the LP's free Margin (Margin [0]).



This is calculated at the bet time, and the LP's capital is moved between margin[0] (free) and margin[1] (locked) depending on whether the bet increases or decreases the LP collective’s net exposure. For example, an initial bet will increase the required margin, but a subsequent small bet on the opposing team would lower the required margin. A bet could move the book so that the net LP liability switches from team 1 to team 0 or consists of the decrease in the net liability on team 1. In any case, the above function captures the difference in the worst-case scenarios for contract liability.

In this way, the LP's total book exposure is cross-margined so that 1.0 AVAX capital can support many bets via incremental bets on both contestants. At settlement, only money not payable is returned to the bookie's free margin pool.

Within the GUI, the maximum bet size is displayed when a user toggles the radio button. It is calculated using the following logic. We use the superscript *i* for the pick, and *-i* for the opponent. The potential liability for pick *i* is



The global maximum exposure for any one match is a function of the amount of LP capital and the concentration factor.



The amount of available LP capital is



With these data, and the odds offered on our pick, we can calculate the maximum exposure for a pick:



To translate this into a betsize, we divide by the payoff odds. For example, if the odds were 1.500, this pays out 50% on each dollar bet (decimal odds -1). Thus with 1.0 in exposure, that would allow a bet for 1/0.5 or 2.0.



LP exposure across matches is independent. The assumption for LP exposure is the worst-case scenario, so there will be no chance of an insolvency, as a bet cannot be taken without capital available.

## Gas for transactions

**contract function gas (x1000)**

oracle initialPost 410

oracle inital voteProcess 129

oracle update Post 112

oracle update voteProcess 72

oracle settle Post 99

oracle settle voteProcess w/ 32 matches 867

oracle settle voteProcess w/ 16 matches 535

oracle settle voteProcess w/ 1 match 191

oracle vote 37

oracle deposit Tokens 52

oracle wd Tokens 84

bet bet 110

bet deposit as LP 54

bet deposit as bettor 48

bet wd as LP 44

bet wd as bettor 33

bet LP claims token Rewards 62

bet bettor redeems bets 55

## Settlement Detail

Settlement records which bets won and then allocates bettor and LP capital to accounts that ensure accrued accounts are fully collateralized. Each bet creates a struct that contains the team and week of the bet. These two inputs create a hash mapped to a number representing its game outcome: 0 for a loss, 1 for a tie, and 2 for a win. When the array of 32 results is sent to the settlement method, the mapping is created (the mapping is zero for uninitialized hashes, so unless updated, the mapping is 0). This mapping is then used for redemptions, in that a bettor claiming his winnings will need the {epoch, match, team} hash to map to a 1 or 2 to generate a payout.

In addition to creating non-zero hash mappings for non-losing teams, the total payments to all bettors were generated using the results and the paySum and betSum arrays:



Here 1*x* is an indicator function that is 1 if true, 0 else. The *WeeklyWinnings* represents the bettor profit, while the WeeklyPayBack represents the initial bettor funding. The oracle fee of 5% is applied to the bettor winnings, representing about 2.5% of the total bet amount. As individual payouts are less than or equal to the total payouts in any week, this rounding truncations on individual redemptions will not compromise contract solvency; rounding will not prevent redemptions.

At settlement, accounts are adjusted as follows:

Redemption capital = WeeklyPayBack

PayoffPot = WeeklyWinnings \* 95 /100

Oracle fee revenue = WeeklyWinnings \* 5 /100

The bookie's capital then adds the money bet that week minus the payouts for wins and ties.

bookiePool = bookiePool + bettorLocked – redemptionPot – payoffPot

The oracle revenues are then just 5% of the WeeklyWinnings, and are transferred to the oracle contract in the settlement function.

The bettor's money exists in the residual and must be claimed via redemption. At redemption, their bet and its winnings are credited to the bettor's user balance, available for withdrawal or future bets.

After settlement, the bookieLocked margin is set to zero, so all LP funds are available for withdrawal. LPs need to pass 2 settlements before they can withdraw.

## Odds Stability

The relative stability of straight-up sporting event odds compared to the vig allows for a super high-latency oracle that would never work for swapping tokens. Consider that your average daily stock price volatility of 2.5% is 16 times greater than the average bid-ask spread of 0.15%. A market maker who adjusted their bid-ask prices once daily would be exposed as a 'money pump' by arbitrageurs, in that if the price moves up 2.5%, the market maker will almost certainly have sold on the way up, generating real-time losses.

In contrast, the implicit spread on money line bets is 2.5% in terms of a win probability. This means one needs a 2.5% edge in predicting which team wins to beat the house. The daily volatility of these odds, when translated into a win probability, is significantly less than half of that, implying the book would make money even if it had day-old odds and the new odds were moving in the right direction. In practice, the closing line and opening line for NFL betting is statistically identical.

Closing line odds are actually worse for bettors than the opening line, though not significantly. Regardless, the risk of arbitrage arises whenever the odds on both sides winning implies a probability of less than 1.0, which allows the bettor to make a certain profit regardless of who wins. If the odds were offered by the same book, and that book allowed margin accounting, this would create a money pump. In practice, the opportunity comes from the odds generated by two separate books, which prevents cross margining these bets. Given there is a limit to any one event, and the opening and closing line odds are statistically equivalent, the LPs would make money even if such an odds discrepancy arose.

1. LP adverse selection: offer price and get filled only when it’s a bad deal, selling when the contract price is low and buying when it is high. [↑](#footnote-ref-1)
2. A flat book does not mean the bet amounts for both teams are *necessarily* equal. For example, a contest could have 5 avax bet on a 5:1 favorite, and 1 avax bet on the underdog, leaving the LPs with zero risk. [↑](#footnote-ref-2)
3. It is equal statistically, that is, over time. Both the oracle and LPs are subject to small sample variation from different sources. If there are no winners, the oracle gets nothing; the LP’s could lose all of the bets where they have net exposure and thus lose money. [↑](#footnote-ref-3)
4. I created something that fills a real niche as opposed to the Ponzi scams that dominate dapps. Online betting, let alone providing liquidity or oracle services, is not legal for me in my juristdiction; with 8 billion people on the planet and such a big market, many will be able and willing. [↑](#footnote-ref-4)
5. I did not create such a vault but it should be straightforward, though there are several ways to do this. [↑](#footnote-ref-5)
6. E.g., prior to commercial civil law there were courts along trade routes throughout Medieval Europe that enforced commercial laws (the *Lex mercatoria*), and its judgments were accepted not out of any legal authority granted by a state's monopoly on violence, but rather refusal would ruin one's business reputation and thus future revenue. [↑](#footnote-ref-6)
7. This can be seen looking at consumer and producer surplus, or the first and second welfare theorems. [↑](#footnote-ref-7)
8. 625 = 50.001% of 1250 [↑](#footnote-ref-8)
9. The bad data sender’s tokens are burned, not redistributed. We do not want to incent the oracle token holders to reject valid submissions. [↑](#footnote-ref-9)
10. The favorite/underdog refers to the opening line, and so over the week the initial favorite may become the underdog. Nonetheless, the ordering is fixed in the initial event posting. [↑](#footnote-ref-10)
11. The initial data provider’s tokens are credited as a yes vote, and votes are decided on a simple majority of votes cast. [↑](#footnote-ref-11)
12. A negative vig would allow someone to create positions that would generate arbitrage profits. [↑](#footnote-ref-12)