A gentle introduction: BOLD

ALPHA RELEASE, PUBLIC PREVIEW DOCS The BOLD dispute protocol is currently deployed on a public testnet and tagged as analpha release. The code has not been audited andshould not be used in production scenarios . Please note that the public testnet is intended for Arbitrum users and researchers to test and experiment with the BOLD dispute protocol for the purposes of education and hardening the protocol via the surfacing of bugs. The public testnet may be paused and its parameters updated at any time in response to scenarios that impact the network and its ability to fulfill its function as a working, reliable test environment. This documentation is currently inpublic preview .

To provide feedback, click theRequest an update button at the top of this documentjoin the Arbitrum Discord , or reach out to our team directly by completingthis form . This introduction is for those who want to learn about BOLD: a new dispute protocol for Optimistic Rollups that can enablepermissionless validation for Arbitrum chains . BOLD stands for Bounded Liquidity Delay and is currently deployed on a public testnet for anyone to join and test how challenges will work.

This next-generation dispute protocol technology will soon be available for any Arbitrum chain, and pending a governance vote, will eventually be made available on Arbitrum Sepolia, Arbitrum One, and Arbitrum Nova.

BOLD will eventually replace the current, permissioned fraud proof mechanism that powers Arbitrum chains today.

In a nutshell:

- Validation for Arbitrum One and Nova is currently is limited to apermissioned set of parties maintained by the Arbitrum DAO
- to reduce the risks ofdelay attacks
-
    - a class of attacks where malicious entities can open many disputes during the challenge period and delay confirmations of assertions by the amount of time needed to resolve those disputes.
- BOLD, an acronym for Bounded Liquidity Delay, is a new challenge resolution protocol for Arbitrum chains that enables permissionless validation by mitigating the risks of delay attacks againstrollups like Arbitrum
- . This is possible because BOLD's design ensures disputes will be resolved within a fixed time window, currently set to equal 1 challenge period (~6.4 days) for Arbitrum One and Nova. If there is a dispute, BOLD guarantees the maximum total time to be equal to 2 challenge periods (1 for raising disputes, 1 for resolving disputes), 1 day grace period for the Security Council to intervene, and a small delta for computing challenges.
- Permissionless validation is a key milestone onArbitrum's journey to becoming a Stage 2 Rollup
-
    - the most advanced and mature rollup technology categorization. With BOLD,any honest party can validate and bond their funds to post a correct L2 state assertions to win disputes against malicious entities.
- BOLD is currently considered to be inalpha
- release and is deployed on a public testnet.Follow this guide
- to deploy a BOLD validator to test and explore, first hand, how BOLD works to secure Arbitrum chains. To learn more about BOLD, please check out theBOLD whitepaper
- andBOLD's code and specifications on Github
- .

Whatexactly

is BOLD?

BOLD, an acronym for Bounded Liquidity Delay Protocol, is an upgrade to Arbitrum's existing dispute protocol. Specifically, BOLD changes some of the rules used by validators to open and resolve disputes about Arbitrum's state to ensure only valid states get confirmed on an Arbitrum chain's parent chain, such as Ethereum.

The current dispute protocol has working fraud proofs and is used in production today by Arbitrum chains. The changes BOLD brings enables anyone to participate in the validation of the state of the chain and enhances security around withdrawals to L1.

A bonded validator's responsibilities are to:

- Post claims about an Arbitrum chain's state to Ethereum,
- Challenge invalid claims made by other validators, and
- Confirm valid claims - either by timing other validators out or by winning a challenge

The goal of BOLD is to unlock permissionless validation by ensuring that disputes are resolved within a fixed period of time (currently equivalent to 2 challenge periods, plus a small grace period for the Security Council and a small delta for computation), effectively removing the risk of delay attacks and making withdrawals to the parent chain more secure. BOLD accomplishes this by introducing a new dispute system that lets any, one entity defend Arbitrum against malicious parties - effectively allowing anyone to validate Arbitrum's state without needing permission to do so.

Why does Arbitrum need a new dispute protocol?

While Arbitrum chains today have working fraud proofs to secure withdrawals, BOLD introduces a few subtle but innovative changes that letanyone challenge and win disputes - all within a fixed time period. In other words, Arbitrum chains will continue to be secured with an interactive proving game between validators, but with the added benefit of this game being completely permissionless and time-bounded to the same length as 1 challenge period (currently set at 6.4 days).

Under the hood, the reason why BOLD can offer time-bound, permissionless validation is because a correct Arbitrum state assertion isnot tied to the entity that bonds their capital to a claim . This property, coupled with the fact that L2 states are completely deterministic and can eventually be proven on Ethereum, means thatany number of honest parties can rely on BOLD to prove that their claim is correct. Lastly, a property that will not change with BOLD is the fact that there needs to only be 1 honest party defending Arbitrum.

BOLD brings Arbitrum closer to being recognized as a Stage 2 rollup

Inspired byVitalik's proposed milestones , the team over at L2Beat has assembled a widely recognized framework for evaluating the development Ethereum Rollups. Both Vitalik and theL2Beat framework refer to the the final stage of rollup development as"Stage 2 - No Training Wheels" . A critical criterion for being considered a Stage 2 rollup is to allow anyone to validate the L2 state and post fraud proofs to Ethereum without restraints. This is considered a key requirement for Stage 2 because it ensures"that the system is not controlled by a limited set of entities and instead is subject to the collective scrutiny of the entire community" .

BOLD enables permissionless validation by allowinganyone to challenge incorrect Arbitrum state assertions and therefore unlocks new avenues for participation in securing the network, fostering greater inclusivity and resilience. This is made possible because BOLD guarantees that a single, honest entity who has their capital bonded to the correct Arbitrum state assertion will always win against malicious adversaries. The research and work to bring BOLD to life underscores Arbitrum's commitment to scaling Ethereum without compromising on security.

With BOLD at its core, Arbitrum charts a course towards being recognized as a Stage 2 rollup by addressing the currently yellow (above) State Validation wedge inL2Beat's risk analysis pie chart . BOLD contributes to a more decentralized, efficient, and robust rollup ecosystem. Additionally, BOLD will be available as an upgrade to all Orbit chains who wish to adopt it to reap the aforementioned benefits.

BOLD makes withdrawals to L1 Ethereum safer

Today, there is a period of time, following a state assertion, called the "challenge period" where any validator can open a dispute on a given state - this is what makes Arbitrum an optimistic rollup. This challenge period is why you must wait ~1 week (6.4 days to be exact) to withdraw assets from Arbitrum One, for example. While this design is secured with working fraud proofs, it is susceptible todelay attacks , where malicious actors continuously open disputes to extend that challenge period for as long as they're willing to sacrifice bonds - effectively extending the challenge period indefinitely by an amount equal to the time it takes to resolve each dispute, one by one. This risk is not ideal nor safe and is why validation for Arbitrum One and Nova is confined to a permissioned set of entities overseen by the Arbitrum DAO.

BOLD addresses these challenges head-on by introducing a time limit on the existing rollup protocol for resolving disputes, effectively ensuring that challenges conclude within a 6.4-day window (this window can changed by the DAO for Arbitrum One and Nova). This is possible due to two reasons: (1) BOLD's design allows for challenges between the honest party and any number of malicious adversaries to happen in parallel, and (2) the use of a time limit that will automatically confirm the honest party's claims if the challenger fails to respond.

To summarize with an analogy and the diagram below: Arbitrum's current dispute protocol assumes that any assertion that gets challenged must be defended against each unique challenger sequentially, like in a"1v1 tournament" . BOLD, on the other hand, enables any single honest party to defend the correct state and be guaranteed to win, similar to an"all-vs-all battle royale" where there must and will always be a single winner in the end.

Note that the timer/clocks above are arbitrary and instead represent the duration of challenges and how challenges are sequential today but can take place in parallel with BOLD. The duration of challenges are independent from one another.

How is this possible?

The BOLD protocol provides the guardrails and rules for how validators challenge claims about the state of an Arbitrum chain. Since Arbitrum's state is deterministic, there will always be only 1 correct state for a given input of on-chain operations and transactions. The beauty of BOLD's design guarantees that disputes will be resolved within a fixed time window, removing the risk of delay attacks and ultimately enabling anyone to bond their funds to and successfully defend that singular correct state of Arbitrum.

Let's dive in to an overview of how BOLD actually works.

1. An assertion is made:

2. Validators begin by taking the most recent confirmedRBlock
3. , calledBlock A
4. , and assert that some number of transactions afterwards, using Nitro's deterministic State Transition Function (STF), will result in an end state,Block Z
5. . If a validator claims that the end state represented byBlock Z
6. is correct, they will bond their funds toBlock Z
7. and propose that state to be posted to Ethereum. If nobody disagrees after a certain amount of time, known as the challenge period, then the state represented by the RBlockBlock Z
8. is confirmed as the correct state of an Arbitrum chain. However, if someone disagrees with the end stateBlock Z
9. , they can submit a challenge. This is where BOLD comes in to play.
10. A challenge is opened:
11. When another validator observes and disagrees with the end state represented byBlock Z
12. , they can permissionlessly open a challenge by asserting and bonding capital to a claim on a different end state, represented by an RBlockBlock Y
13. . At this point in time, there are now 2 asserted states:Block A → Block Z
14. andBlock A → Block Y
15. . Each of these asserted states, at this point in time now that there's a challenge, are referred toedges
16. while a Merkle tree of asserted states from some start to end point (e.g.Block A → Block Z
17.) is more formally known as ahistory commitment.
18. It is important to note that Ethereum at this point in time has no notion of which edge(s) is correct or incorrect - edges are simply a portion of a claim made by a validator about the history of the chain from some end state all the way back to some initial state. Also note that because a bond put up by a validator is tied to an assertion rather than the party who put up that bond, there can be any number of honest, anonymous parties that can open challenges against incorrect claims. It is important to note that the bonds put up to open challenges are held in a Gnosis Safe multi-sig wallet controlled by the Arbitrum Foundation.
19. Multi-level, interactive dissection begins:
20. To resolve the dispute, the disagreeing entities will need to come to an agreement on what theactual, correct
21. asserted state should be.It would be tremendously expensive to re-execute
22. and compare everything fromBlock A → Block Z
23. andBlock A → Block Y
24. , especially since there could be potentially millions of transactions in betweenA
25. ,Z
26. , andY
27. . Instead, entities take turns bisecting their respectivehistory commitments
28. until they arrive at a single step of instruction where an arbiter, like Ethereum, can declare a winner. Note thatthis system is very similar to how challenges are resolved on Arbitrum chains today
29.
    - BOLD only changes some minor, but important, details in the resolution process. Let's dive into what happens next:1. Block challenges
30.
    1. : when a challenge is opened, the edges are calledlevel-zero edges
31.
    1. since they are at the granularity of Arbitrum blocks. The disputing parties take turns bisecting their history commitments until they identify the specific block that they disagree on.
32.
    1. Big-step challenges:
33.
    1. now that the parties have narrowed down their dispute to a single block, that we callBlock B
34.
    1. , the back-and-forth bisection exercise continues within that block. Note thatBlock B
35.
    1. is claimed by all parties to be some state after the initial stateBlock A
36.
    1. but before the final statesBlock Z
37.
    1. andBlock Y
38.
    1. . This time however, the parties will narrow down on a specificrange
39.
    1. of instructions for the state transition function within the block - essentially working towards identifying a set of instructions that their disagreement lies within. This range is currently defined to be 2^20 steps of WASM instructions, which is the assembly of choice for validating Arbitrum chains.
40.
    1. One-step challenge:
41.
    1. within that range of 2^20 instructions, the back and forth bisecting continues until all parties arrive at a single step of instruction that they disagree on. At this point in time, parties agree on the initial state of Arbitrum before the step but disagree on the end state 1 step immediately after. Remember that since Arbitrum's state is entirely

deterministic, there is only 1 correct end state.

42. One-step proof:
43. Once a challenge is isolated down to a dispute about a single step, both parties run that step to produce, and then submit, a one-step proof to the OneStepProof smart contract on the parent chain (e.g. Ethereum). A one-step proof is a proof that a single step of computation results in a particular state. The smart contract on the parent chain will execute the disputed step to validate the correctness of a submitted proof from the two parties. It is at this point that the honest party's proof will be deemed valid and it's tree of edges will be confirmable by time, while the dishonest party has their edges rejected.
44. Confirmation:
45. Once the honest one-step edge is confirmed, the protocol will work on confirming or rejecting the parent edges until it reaches the level-zero edge of the honest party. With the honest party's level-zero edge now confirmed, the honest party's assertion bond can be withdrawn. Meanwhile, the dishonest party has their bonds taken away to ensure the dishonest party is always punished. Reimbursements for the honest party's L1 gas costs and mini-bonds made at the other challenge levels are handled by the Arbitrum Foundation.1. There is another way that a level-zero edge can get confirmed: time. At each of the mini-stages of challenge (block challenge, big-step challenge, one-step challenge), there is a timer that increments upwards towards some challenge period,T
46.
    1. defined by BOLD. This timer begins ticking for a party when they submit their bisected history commitment until their challenger submits their bisected history commitment in response. An edge is automatically confirmed if the timer reachesT.

That's it! We've now walked through each of the steps that validators will take to dispute challenges with the BOLD protocol. One final note here is that each of the steps explained above can take place concurrently and this is one of the reasons why BOLD can guarantee that disputes are resolved within a fixed time frame.

BOLD's economics and spam protection

Given that participation in BOLD is permissionless, it is recommended that the size of the bonds required to participate be high enough to disincentivize malicious actors from attacking Arbitrum One and Nova and to mitigate against spam (that would otherwise delay confirmations). High bonding values do not harm decentralization because (1) trustless bonding pools can be deployed permissionlessly to let the community open challenges and post assertions, and (2) any number of honest parties of unknown identities can emerge to bond their funds to the correct assertion and participate in the defense of Arbitrum at any time within a challenge. As with the current dispute resolution protocol, there are no protocol level incentives for parties who opt in to participate in validating Arbitrum One and Nova with BOLD. The bonds can be any ERC20 token and be set to any size for Arbitrum One and Nova, as determined by the ArbitrumDAO.

The following link,Economics of Disputes in Arbitrum BOLD, covers the rationale behind the design and recommended values for the bonds. Note that the ArbitrumDAO can change these values and the type of asset used for the bonds via a governance proposal.

BOLD makes permissionless validation possible for Arbitrum rollup chains and marks a major step towardsfull decentralization . This significant milestone also lays the groundwork for productive discussions about future economic incentives for those participating in the protocol since anyone can participate.

Reimbursements and penalities

As mentioned above in Step 5, once all of a validator's assertions are confirmed, a validator can withdraw their full assertion bond. Other costs, including the mini-bonds from both the malicious actor and honest party, will continue to be held in the Gnosis Safe multi-sig wallet controlled by the Arbitrum Foundation.

The ArbitrumDAO has full discretion over:

- How to reimburse the mini-bond and gas costs to honest parties, and
- What to do with the funds confiscated from a malicious actor (including, but not limited to, rewarding the honest parties with a portion of the confiscated funds, burning the confiscated funds in its entirety, or sending the confiscated funds to the DAO treasury).

Note that honest parties are not automatically rewarded with the funds confiscated from malicious actors to avoid creating a situation where honest parties wastefully compete to be the first one to make each honest move in the interactive fraud proof game. Additionally, BOLD resolves disputes by determining which top-level assertion is correct, without necessarily being able to classify every move as "honest" or "malicious" as part of the interactive fraud proof game without off-chain knowledge. These two factors are the reason why the ArbitrumDAO has the final authority on what to do with the funds that are held in Gnosis Safe multi-sig wallet.

What can I do with BOLD today?

Today, BOLD is deployed on a public testnet using Ethereum Sepolia as a base layer for anyone to experiment with and test on. The intent behind this testnet is purely to demonstrate, first-hand, how disputes can effectively resolved by a single party in a fixed challenge period on Arbitrum chains. Feedback gained from developers, users, and researchers will help improve

and strengthen BOLD's design.

If you're intrigued by what BOLD can unlock for Arbitrum chains, we encourage you to interact with BOLD by:

- Following this guide
- to deploy a BOLD validator to test and explore, first hand, how BOLD works to secure Arbitrum chains. A BOLD testnet block explorer
- is also available for you to peruse!
- Checking out this BOLD Technical Deep Dive
- to learn about BOLD's implementation, alongside the BOLD source code on Github
- to understand how BOLD works under the hood.
- Reviewing the Economics of Disputes in Arbitrum BOLD
- to learn about rationale behind the design and recommended bonding values.
- Reading the formal specification and mathemtical proofs behind BOLD in the BOLD whitepaper
- .

Wen mainnet?

BOLD is inalpha , which means there are a lot of planned improvements on the roadmap. A few high-level next steps for BOLD's journey to being deployed to Arbitrum chains include:

- A comprehensive, third-party audit of the BOLD source code
- to ensure the effectiveness and safety of the design.
- Tools and frameworks for the smooth migration of existing validators and a seamless on-boarding for new validators to use BOLD for their respective Arbitrum chains.
- Monitoring stack for people to use to see on-going challenges on the testnet
- A mechanism for the community to pool funds together to bond capital to an assertions made by validators
- The launch of a public bounty program for white hat auditors and security professionals to help test and secure the BOLD protocol design.
- Proposing, to the Arbitrum DAO, that the BOLD protocol be adopted - first for Arbitrum Sepolia and then eventually for Arbitrum One and Arbitrum Nova.
- Cutting a GA release of Nitro that enables BOLD validation.

Frequently asked questions about BOLD (FAQ):

How does bonding work?

- The entities responsible for posting assertions about Arbitrum state to Ethereum are called validators. If posting assertions were free, anyone could create conflicting assertions to always delay withdrawals by 14 days instead of 7. As such, Arbitrum requires validators to put in a "security deposit", known as a bond, to be allowed to post assertions. Validators can withdraw their bond as soon as their latest posted assertion has been confirmed, and end their responsibilities. These bonds can be any ERC20 token and should be set to a large enough value (e.g. 200 WETH) to make it economically infeasible for an advesary to attack an Arbitrum chain and to mitigate against spam (that would otherwise delay confirmations). Requiring a high bond to post assertions about Arbitrum seems centralizing, as we are replacing a whitelist of validators with instead a system that requires a lot of money to participate in. To address this, there is a contract
- that anyone can use to deploy a bonding pool as a way of crowdsourcing funds from others who wish to help defend Arbitrum but who may not individually be able to put up the large upfront bond itself. The use of bonding pools, coupled with the fact that there can be any number of honest anonymous parties ready to defend Arbitrum, means that these high bond values do not harm decenentralization.

What is the user flow for using the assertion bonding pool contract?

- Anyone can deploy an assertion bonding pool using AssertionStakingPoolCreator.sol
- as a means to crowdsource funds to put up a bond for an assertion. To defend Arbitrum using a bonding pool, an entity would first deploy this pool with the assertion that they believe is correct and wish to put up a bond to challenge an advesary's assertion. Then, anyone can verify that the claimed assertion is correct by running the inputs through their node's State Transition Function (STF). If other parties agree on the assertion being correct, then they can deposit their funds into the contract. When enough funds have been deposited, anyone can permissionlessly trigger the creation of the assertion on-chain to start the challenge. Finally, once the honest parties' assertion is confirmed by the dispute protocol, all involved entities can get their funds reimbursed and can withdraw.

Are there any incentives to run a BOLD validator to secure Arbitrum chains?

- Running a BOLD validator secures their respective Arbitrum chain and protects the assets on the chain from malicious actors - all you need is 1 honest party. Other than this critical piece, there are currently no financial incentives for parties to run a BOLD validator. Any future decisions or changes to this design can be proposed to and voted on by the Arbitrum DAO.

What type of hardware will be necessary to run a BOLD validator?

- The minimum hardware requirements for running a BOLD validator is still being researched and finalized. The goal, however, is that regular consumer hardware (i.e. laptop) can effectively be used by an honest party to secure an Arbitrum chain using BOLD in the average case.

How do BOLD validators communicate with one another? Is it over a P2P network?

- BOLD validators for Arbitrum chains communicate directly with smart contracts on L1 Ethereum. This means that opening challenges, submitting bisected history commitments, one-step proofs, and confirmations are all refereed on Ethereum. There is no p2p between validators.

For an L3 Orbit chain, secured using BOLD, that settles to Arbitrum One, does the one-step proof happen on Arbitrum One?

- Yes

Does implementing BOLD reduce the scope or remove the need for the Arbitrum Security Council?

- BOLD can limit the scope of Arbitrum One and Nova's reliance on the Security Council as it takes Arbitrum chains one-step closer to full decentralization. Edit this page Last updatedonApr 18, 2024Previous WAVM Modules Next Public preview