

Truffle and Infura Now Support Arbitrum

In our continuing efforts to help build the emerging multi-network Ethereum ecosystem, the Truffle team is thrilled to announce the launch of the Arbitrum Truffle Box. The usability and efficiency of Ethereum depends on scalability, and Arbitrum's Layer 2 solution blends security and higher throughput to help make scalable Ethereum a reality.

Arbitrum

Arbitrum is a Layer 2 scaling solution for Ethereum that uses optimistic rollups with interactive proving to make smart contracts scalable, fast, and private. It tracks closely to Ethereum, so that Ethereum developers can easily cross-compile their contracts to deploy on Arbitrum. Arbitrum offers its developers a variety of benefits: trustless security, compatibility with Ethereum, scalability, and cost efficiency.

The Arbitrum rollup chain is built on top of and secured by Ethereum. From the developer perspective, interacting with Arbitrum is nearly identical to interacting with Ethereum. Arbitrum supports the same RPC interface as Ethereum, supports all EVM languages, and natively supports all Ethereum tooling without any special adapters. The main difference users and developers will notice is that transactions on Arbitrum cost a fraction of what they would if run natively on Ethereum.

It is important to understand the mechanics of withdrawing funds from the Arbitrum chain. When withdrawing to Ethereum through an Arbitrum bridge, there is a delay to ensure that the transactions being rolled up are confirmed. Information about this finality delay and potential methods to avoid it can be found in Arbitrum's documentation [here](#).

For more details on Arbitrum's protocol, see their [complete documentation](#).

Introducing The Arbitrum Truffle Box

The [Truffle Arbitrum Box](#) provides developers with the boilerplate structure necessary to start coding for Arbitrum's Ethereum Layer 2 solution.

We provide the initial building blocks to get started working on Arbitrum without pushing developers to write any particular sort of application. With this box, you will be able to compile, migrate, and test Solidity code against several instances of Arbitrum Networks.

Arbitrum's Layer 2 solution is almost fully compatible with the EVM. The main difference between the EVM and the Arbitrum chain that developers will notice is that some opcodes are different and concepts such as time and gas are handled a little differently. Developers can use their regular Solidity compiler to compile contracts for Arbitrum. You can see the complete list of differences between the Arbitrum L2 chain and Ethereum [here](#).

Integration With Infura

Our colleagues at Infura are also adding full API support for the Arbitrum network. Infura users can now add the free Arbitrum add-on directly in their Infura dashboard. The Arbitrum network API largely uses the same JSON RPC specification as Ethereum and Infura will support all the necessary API methods for building a fully functional application on the Arbitrum network. For more information on the full support check out the [complete documentation](#).

Infura endpoints for the Arbitrum chain are included in the configuration of the Arbitrum Truffle Box, to aid developers in getting fully acquainted with their options when developing for Arbitrum.

Where To Get Help

You can find all of the documentation for the Truffle Arbitrum Box [here](#). If you have any questions, feedback, or just want to chat, reach out to us through the resources on our [community page](#). We are thrilled to add Arbitrum to the Truffle development suite and can't wait to see the universe of applications built on Layer 2. Stay tuned as we continue to add more networks, features, and tools that accelerate and simplify development on a secure and scalable Ethereum.