

Derivation Hacks

⚠ OP Stack Hacks are explicitly things that you can do with the OP Stack that are not currently intended for production use.

OP Stack Hacks are not for the faint of heart. You will not be able to receive significant developer support for OP Stack Hacks — be prepared to get your hands dirty and to work without support.

Overview

This guide teaches you how to modify the default Derivation layer module for an OP Stack chain. The Derivation layer is responsible for parsing the raw inputs from the Data Availability layer and converting them into [Engine API \(opens in a new tab\)](#) payloads to be sent to the Execution layer. The Derivation Layer is generally tightly coupled to the Data Availability layer because it must understand both the APIs for the Data Availability layer module(s) of choice and the format of the raw data published to the chosen module(s).

Default

The default Derivation layer module is the Rollup module. This module derives transactions from three sources: Sequencer transactions, user deposits, and L1 blocks. The Rollup module also enforces certain ordering properties that, for example, guarantee that user deposits are always included in the L2 chain within a certain configurable amount of time.

Security

Modifying the Derivation layer can have unintended consequences. For example, removing or extending the time window in which user deposits must be included can allow a Sequencer to censor the L2 chain. Because of the flexibility of the Derivation layer, the exact impact of any change is likely to be unique to the specifics of the change. The negative impacts of any modifications should be carefully considered on a case-by-case basis.

Modding

EVM Event-Triggered Transactions

The default Rollup configuration of the OP Stack includes "deposited" transactions that are triggered whenever a specific event is emitted by the OptimismPortal contract on L1. Using the same principle, an OP Stack chain can derive transactions from events emitted by any contract on an EVM-based DA. Refer to [attributes.go \(opens in a new tab\)](#) to understand how deposited transactions are derived and how custom transactions can be created.

EVM Block-Triggered Transactions

Like with events, transactions on an OP Stack chain can be triggered whenever a new block is published on an EVM-based DA. The default Rollup configuration of the OP Stack already includes a block-triggered transaction in the form of [the "L1 info" transaction \(opens in a new tab\)](#) that relays information like the latest block hash, timestamp, and base fee into L2. The Getting Started guide demonstrates the addition of a new block-triggered transaction in the form of a new transaction that reports the amount of gas burned via the base fee on L1.

And much, much more...

The Derivation layer is one of the most flexible layers of the stack. Transactions can be generated from all sorts of raw input data and can be triggered from all sorts of conditions. You can derive transactions from any piece of data that can be found in the Data Availability layer modules!

[Tutorial: Adding attributes to the derivation function](#).

[Data Availability Hacks](#) [Execution Hacks](#)