Smart Contract Overview

This guide explains how smart contracts work at a high level.

L1 contracts

L2OutputOracle

TheL2OutputOracle contract(opens in a new tab) contains the state root of the Optimism blockchain (OP Mainnet, OP Sepolia, etc.). Once fault proofs are activated, it will be the one that receives the result of the fault proof process.

This is the contract that replaces the old State Commitment Chain.

OptimismPortal

TheOptimismPortal contract(opens in a new tab) provides the low-level API for communications between layers. Unless you are trying to send L2 transactions via L1 to bypass the sequencer, we strongly recommend sending messages between L1 and L2 via the L1CrossDomainMessenger and L2CrossDomainMessenger.

L1CrossDomainMessenger

TheL1CrossDomainMessenger contract(opens in a new tab) is used for sending messages between the underlying L1 (Ethereum, Sepolia, etc.) and L2 (OP Mainnet, OP Sepolia, etc.).

L1StandardBridge

TheL1StandardBridge contract(opens in a new tab) usesL1CrossDomainMessenger to transfer ETH and ERC-20 tokens between the underlying L1 (Ethereum, Sepolia, etc.) and L2 (OP Mainnet, OP Sepolia, etc.).

L2 contracts (predeploys)

L1Block

TheL1Block contract(opens in a new tab) sits at address0x4200000000000000000000000000000000000015 . You can usethe getter functions(opens in a new tab) to get these parameters:

- number
- : The latest L1 block number known to L2 (theL1BlockNumber
- contract is still supported to avoid breaking existing applications)
- timestamp
- : The timestamp of the latest L1 block
- basefee
- : The base fee of the latest L1 block
- hash
- : The hash of the latest L1 block
- sequenceNumber
- : The number of the L2 block within the epoch (the epoch changes when there is a new L1 block)

Currently the L1 information is delayed by two block confirmations (~24 seconds) to minimize the impact of reorgs.

SequencerFeeVault

TheSequencerFeeVault contract(opens in a new tab) handles funding the sequencer on L1 using the ETH base fee on L2.

The fees are calculated usingEIP 1559(opens in a new tab) , the same mechanism that Ethereum uses (but with different parameter values).

L2ToL1MessagePasser

TheL2ToL1MessagePasser contract(opens in a new tab) is used internally byL2CrossDomainMessenger to initiate withdrawals.

Note that there are two contracts under this name:

- The legacy contract(opens in a new tab)
- at address0x4200000000000000000000000000000000000000

- The new contract(opens in a new tab)
- at address0x4200000000000000000000000000000000000016

L2CrossDomainMessenger

TheL2CrossDomainMessenger contract(opens in a new tab) is used to send messages from L2 (OP Mainnet, OP Sepolia, etc.) to the underlying L1 (Ethereum, Sepolia, etc.).

L2StandardBridge

TheL2StandardBridge contract(opens in a new tab) is a wrapper around theL2CrossDomainMessenger that can handle token transfers.

WETH9

The WETH9 contract(opens in a new tab) is an ERC-20 token that wraps around ETH to provide extra functionality, such as approvals.

Legacy Contracts

Those are contracts that have been superseded, but are kept in case any deployed contract depends on them.

- L1BlockNumber(opens in a new tab)
- :
- TheL1BlockNumber
- contract provides the number of the latest L1 block.
- In Bedrock it is simply a proxy toL1Block
- .
- DeployerWhitelist(opens in a new tab)
- :
- TheDeployerWhitelist
- contract used to manage the whitelist beforeOP Mainnet moved out of beta(opens in a new tab)
- .
- OVM_ETH(opens in a new tab)
- :
- TheOVM_ETH
- contract used to manage users ETH balances prior to Bedrock.

Design Philosophy & Principles Deposit Flow