Difference with Ethereum

While Filecoin EVM runtime aims to be compatible with the Ethereum ecosystem, it has some marked differences.

Gas costs

Filecoin charges Filecoin gas only. This includes the Filecoin EVM runtime. Instead of the Filecoin EVM runtime charging gas according to the EVM spec for each EVM opcode executed, the Filecoin virtual machine (FVM) charges Filecoin gas for executing the EVM interpreter itself. TheHow gas works page goes into this in more detail. Importantly, this means that Filecoin EVM runtime gas costs and EVM gas costs will be very different:

1. EVM and Filecoin gas are different units of measurement and are not 1:1. Purely based on chain throughput (gas/second), the ratio of Ethereum gas to Filecoin gas is about 1:444. Expect Filecoin gas numbers to lookmuch
2. larger than those in Ethereum.
3. Because Filecoin charges Filecoin gas for executing the Filecoin EVM runtime interpreter:
4.
    1. Some instructions may be more expensive and/or cheaper in Filecoin EVM runtime than they are in the EVM.
5.
    1. EVM instruction costs can depend on the exact Filecoin EVM runtime code-paths taken, and caching.
6. 3.
7.

⚠ Filecoin gas costs are not set in stone and should never be hard-coded. Future network upgrades will break any smart contracts that depend on gas costs not changing.

Gas stipend

Solidity callsaddress.transfer andaddress.send to grant a fixed gas stipend of 2300 Ethereum gas to the called contract. The Filecoin EVM runtime automatically detects such calls, and sets the gas limit to 10 million Filecoin gas. This is a relatively more generous limit than Ethereum's, but it's future-proof. You should expect the address called to be able to carry out more work than in Ethereum.

Self destruct

Filecoin EVM runtime emulates EVM self-destruct behavior but isn't able to entirely duplicate it:

1. There is no gas refund for self-destruct.
2. On self-destruct, the contract is marked as self-destructed, but is not actually deleted from the Filecoin state-tree. Instead, it simply behaves as if it does not exist. It acts like an empty contract.
3. Unlike in the EVM, in Filecoin EVM runtime, self-destruct canfail
4. causing the executing contract to revert. Specifically, this can happen if the specified beneficiary address is an embeddedID address
5. and no actor exists with the specified ID.
6. If funds are sent to a self-destructed contract after it self-destructs but before the end of the transaction, those funds remain with the self-destructed contract. In Ethereum, these funds would vanish after the transaction finishes executing.
7.

CALLCODE

TheCALLCODE opcode has not been implemented. Use the newerDELEGATECALL opcode.

Bare-value sends

In Ethereum,SELFDESTRUCT is the only way to send funds to a smart contract without giving the target smart contract a chance to execute code.

In Filecoin, any actor can usemethod 0 , also called a bare-value send, to transfer funds to any other actor without invoking the target actor's code. You can think of this behavior as having the suggestedPAY opcode already implemented in Filecoin.

Precompiles

The Filecoin EVM runtime, unlike Ethereum, does not usually enforce gas limits when calling precompiles. This means that it isn't possible to prevent a precompile from consuming all remaining gas. Thecall actor andcall actor id precompiles are the exception. However, they apply the passed gas limit to the actor call, not the entire precompile operation (i.e., the full precompile execution end-to-end can use more gas than specified, it's only the finalsend to the target actor that will be limited).

Multiple Addresses

In Filecoin, contracts generally have multiple addresses. Two of these address types,f0 andf410f , can be converted to 0x-style (Ethereum) addresses which can be used in theCALL opcode. SeeConverting to a 0x-style address for details on how these addresses are derived.

Importantly, this means that any contract can be called by either its “normal” EVM address (corresponding to the contract’sf410f address) or its “masked ID address” (corresponding from the contract’sf0 address).

However, the addresses returned by the CALLER, ORIGIN, and ADDRESS instructions will always be the same for the same contract.

- The ADDRESS will always be derived from the executing contract’sf410f
- address, even if the contract was called via a masked ID address.
- The CALLER/ORIGIN will be derived from the caller/origin’sf410f
- address, if the caller/origin is an Ethereum-style account or an EVM smart contract. Otherwise, the caller/origin’s “masked ID address” (derived from theirf0
- address) will be used.
-

Deferred execution model

When calling an Ethereum method that allows the user to ask for thelatest block, Filecoin will return thechain head -1 block. This behavior was implemented for compatibility with the deferred execution mode that Filecoin uses. In this mode, messages submitted at a givenheight are only processed atheight +1 . This means that receipts for a block produced atheight are only available atheight +1 .

Previous FILForwarder Next How gas works

Last updated7 months ago