IRouterClient API Reference

Add Chainlink CCIP to your project

If you need to integrate Chainlink CCIP into your project, install the@chainlink/contracts-ccip NPM package .

npmyarnIf you useNPM :

npminstall@chainlink/contracts-ccip--saveIf you useYarn :

yarnadd@chainlink/contracts-ccip

To send messages through CCIP, users must interact with theIRouterClientinterface. After you importIRouterClient.sol, you can initialize a router client instance:

import{IRouterClient}from"@chainlink/contracts-ccip/src/v0.8/ccip/interfaces/IRouterClient.sol";...IRouterClient router;constructor(address_router){router=IRouterClient(_router);}

Errors

UnsupportedDestinationChain

errorUnsupportedDestinationChain(uint64destChainSelector)

InsufficientFeeTokenAmount

errorInsufficientFeeTokenAmount()

InvalidMsgValue

errorInvalidMsgValue()

Functions

isChainSupported

functionisChainSupported(uint64chainSelector)externalviewreturns(boolsupported) Checks if the given chain ID is supported for sending/receiving.

Parameters

NameTypeDescriptionchainSelectoruint64The chain to check.

Return Values

NameTypeDescriptionsupportedboolis true if supported or false if not.

getSupportedTokens

functiongetSupportedTokens(uint64chainSelector)externalviewreturns(address[]tokens) Gets a list of all supported tokens which can be sent or received to or from a given chain ID.

Parameters

NameTypeDescriptionchainSelectoruint64The chainSelector.

Return Values

NameTypeDescriptiontokensaddress[]The addresses of all supported tokens.

getFee

functiongetFee(uint64destinationChainSelector,structClient.EVM2AnyMessage message)externalviewreturns(uint256fee) returns 0 fees on invalid message.

Parameters

NameTypeDescriptiondestinationChainSelectoruint64The destination chainSelectormessagestructClient.EVM2AnyMessageThe cross-chain CCIP message, including data and/or tokens

Return Values

NameTypeDescriptionfeeuint256returns guaranteed execution fee for the specified message delivery to the destination chain

ccipSend

functionccipSend(uint64destinationChainSelector,structClient.EVM2AnyMessage message)externalpayablereturns(bytes32)
Request a message to be sent to the destination chain.

caution

If themsg.valueexceeds the required fee fromgetFee, the over overpayment is accepted with no refund.

Parameters

NameTypeDescriptiondestinationChainSelectoruint64The destination chain IDmessagestructClient.EVM2AnyMessageThe cross-chain CCIP message, including data and/or tokens

Return Values

NameTypeDescription[0]bytes32messageId The message ID