Just-In-Time Gas

How to use just-in-time gas to execute cross-chain transactions

There are some actions and contracts where cross-chain execution with a relayer doesn't work. For example, most 'sell' actions depend onmsg.sender

- Selling a nft
- Swapping an erc20
- Bidding in an auction

To get around this, you can simply bridge a small amount of ETH for gas and have the user execute the transaction themselves. This is only now possible because of how fast and cheap Relay makes bridging.

We will soon add functionality to our SDK to handle this flow completely, but until then, this guide will demonstrate how you could implement this ux yourself.

Overview of steps:

- **Step 1: Estimate gas needed on the Destination Chain**
- **Step 2: Bridge Just-In-Time gas money from Origin to Destination chain**
- **Step 3: Execute transaction/s on the Destination chain**
- **Step 4 (optional): Bridge back proceeds to Origin chain**

The following example assumes you are set up with theRelay SDK as well asviem . You can skip ahead to the full code and live examplehere .

Step 1: Estimate gas needed on the Destination Chain

Copy const estimatedGas= await publicClient. estimateContractGas ({ ... wethContract, functionName: 'withdraw' , args: [wethBalance] , account: address, }) const gasPrice= await publicClient. getGasPrice ( ) const totalGasEstimation= estimatedGas *gasPriceconst totalGasEstimationWithBuffer= totalGasEstimation+ (totalGasEstimation*BigInt (5)) / BigInt (100) // add 5% buffer to handle gas fluctuation We start by using viem'sestimateContractGas to calulate the gas required to successfully execute the transaction we want to make on the Destination Chain. We also usegetGasPrice to get the current price of gas on the Destination chain. We multiply these two values together to get the total gas estimate and add a 5% buffer to handle any flucuation.

Step 2: Bridge Just-In-Time gas money from Origin to Destination chain

Copy await relayClient. actions. bridge ({ chainId: baseSepolia. id, toChainId: sepolia. id, wallet, value: totalGasEstimationWithBuffer. toString ( ) , onProgress (steps) { setStep (getCurrentStepDescription (steps)) } , }) We use the Relay SDK'sbridge action to bridge the gas money from the Origin to the Destination chain.

Step 3: Execute transaction/s on the Destination chain

Copy const { request} = await publicClient. simulateContract ({ ... wethContract, account: address, functionName: 'withdraw' , args: [wethBalance] , chain: sepolia, gas: estimatedGas, }) const hash= await wallet. writeContract (request) await publicClient. waitForTransactionReceipt ({ hash, }) In this example wewithdraw some WETH with viem'swriteContract and wait for the transaction to complete withwaitForTransactionReceipt .

Step 4 (optional): Bridge back proceeds to Origin chain

Copy const destinationEthBalance= await getBalance (wagmiConfig, { address: address, chainId: sepolia. id, }) const { fees} = (await relayClient. actions. bridge ({ chainId: sepolia. id, toChainId: baseSepolia. id, wallet, value:

destinationEthBalance. value. toString ( ) , precheck: true , // when enabled, skips executing the steps })) as Executeconst bufferedGasFee= BigInt (fees?. gas?? 0) * (BigInt (100) + BigInt (5)) / BigInt (100) // add 5% buffer to handle flucation const amountToBridgeBack= destinationEthBalance. value- bufferedGasFee- BigInt (fees?. relayer?? 0) await relayClient. actions. bridge ({ chainId: sepolia. id, toChainId: baseSepolia. id, wallet, value: amountToBridgeBack. toString ( ) , onProgress (steps) { setStep (getCurrentStepDescription (steps)) } , }) This step is optional and the amount you bridge back would depend on your use case. In this example, we attempt to bridge back all of the ETH that the user has on the Destination chain. We first fetch the user's ETH balance on Destination. We then calculate the required fees in order to bridge that ETH back (gas fee + relayer fee). Once we subtract those fees from the user's Destination ETH balance, we get the total amount that we can afford to bridge and we can execute the bridge action.

Example

Check out thiscodesandbox to see a live example of just-in-time bridging.UX Best Practices Deep Linking twitter Powered by Mintlify