

Cross-Chain Execution

Execute cross-chain transactions via Relay

Relay enables cross-chain execution for arbitrary cross-chain transactions, as well as being integrated with Reservoir's pre-defined NFT actions (buy, sell, & mint). Using Relay, your users can pay for any action on any supported chain, by simply submitting the destination chain call data to Relay.

How it Works

Relay's cross-chain execution is powered by cross-chain relaying. This model results in low-cost, low-latency (1-10 seconds) cross-chain transactions. When a user wishes to perform a cross-chain action, they submit the arbitrary call data to get a quote from relayers for execution. When they accept the quote, the order is validated and they submit a transfer directly to the relayer. The relayer then executes the calldata on the destination chain.

While the Relay Protocol is still in development, users rely on [Reservoir](#) for order validation and relaying. Applications trust Reservoir to perform cross-chain relaying, and refund money in the case of disputes. The team will be releasing a fully decentralized and trustless protocol shortly, which you can learn more about in the [Relay Protocol Documentation](#).

Using Relay's Cross-chain Execution

You can use Relay Cross-chain Execution with custom call methods or with the pre-defined methods of the Reservoir Aggregator.

Custom Call Methods

Relay lets you execute arbitrary call data cross-chain using a custom call method. You can call custom contract methods, mint from custom contracts, mint + bridge (mint an NFT and bridge additional funds in one), or any other custom call execution.

To get started make sure that you install and configure the [Relay SDK](#). Ensure that everything is configured properly before moving on to implementing the call action.

Examples

Here we consider 4 generic executions using Relay.

- Bridge
- : Bridge funds to another chain
- Mint + Bridge
- : Mint an NFT on another chain and bridge additional funds in a single call
- Custom Mint
- : Mint an NFT from a custom contract on any supported chain
- Custom Contract Call
- : Call a non-NFT related contract method

Below are examples of how you could leverage the flexibility of the [call](#) action using the Relay SDK.

```
Instant Bridge Non-NFT action Mint + Bridge Custom Mint Copy
import { getClient, Execute } from "@reservoir0x/relay-sdk";
import { createWalletClient, http } from 'viem' ... wallet= createWalletClient ( { account: address, transport: http ( ) } ) //A
simple bridge requires no data, so 0x (null data) is provided //In this example we're bridging from zora to base
getClient ( ) .
actions . call ( { chainId: 7777777 , toChainId: 8453 , txs: [ { to: "0x03508bB71268BBA25ECaCC8F620e01866650532c" ,
value: "1000000" , } ] , wallet: wallet, onProgress : ( steps ) => { console . log ( steps ) } , } )
```

Cross-chain Execution with the Reservoir Aggregator

To use cross-chain execution with the Reservoir Aggregator, i.e. to mint, buy or sell NFTs aggregated by Reservoir, simply use the Reservoir SDK to trigger cross-chain execution with the appropriate method. When using the Reservoir SDK to

trigger a cross-chain purchase all that's required is additionally passing `currencyChainId` to the `buyToken` method. This assumes you know which token you would like to buy or mint. You can leverage the Reservoir API to look at aggregated collection liquidity, or trending mints. Below is an example of what that would look like to buy a token from secondary:

```
Cross-chain Example Copy import { getClient, Execute } from "@reservoir0x/reservoir-sdk" ; import { createWalletClient, http } from 'viem' ... address= "0x8ba1f109551bD432803012645Ac136ddd6000000" wallet= createWalletClient ( { account: address, transport: http ( ) } ) getClient ( ) ?. actions. buyToken ( { items: [ { token: "0xbc4ca0eda7647a8ab7c2061c2e118a18a936f13d:1" , quantity: 1 } ] , options: { currencyChainId: 10 } , wallet, onProgress: ( steps: Execute[ 'steps' ] ) => { console . log ( steps ) } } ) ` Instant Bridging Contract Compatibility twitter  
Powered by Mintlify
```

- [How it Works](#)
- [Using Relay's Cross-chain Execution](#)
- [Custom Call Methods](#)
- [Examples](#)
- [Cross-chain Execution with the Reservoir Aggregator](#)