

Overview

Summary

Wormhole Connect is a React widget that lets developers offer easy, customized access to Wormhole powered bridges directly in a web application. Connect supports multiple forms of bridging including native asset bridge, Portal wrapped asset bridge, CCTP USDC bridge, and many others. Connect augments each bridge with gas dropoff (a transaction that leaves a user with extra native token so they can pay gas for subsequent on chain interactions) and gasless transactions (Connect relayers pay gas on behalf of users).

Check out the [Github repository](#) !

Wormhole Connect Screenshot The [Wormhole Typescript SDK](#) allows you to implement the same functionality as the Connect widget, but in your own UI. For more information on using the SDK instead of Connect [check out the docs](#) .

Demo

Wormhole Connect is deployed live in several production apps. Here are a few:

- [Portal Bridge](#)
- [Jupiter](#)
- [Pancake Swap](#)
-

Functionality in Connect

1. Gasless Txns:
2. Users can bridge without paying gas on the destination chain.
3.
 1. ETH on Ethereum → whETH on Solana, only pays gas in ETH on Ethereum (covers both send and redeem)
4. 2.
5. Gas Dropoff:
6. User can bridge while only paying gas on the source chain and swap some of the transferred asset into the destination gas token.
7.
 1. ETH on Ethereum → whETH on Avalanche + AVAX on Avalanche, only pays gas in ETH on Ethereum (covers both send and redeem)
8. 2.
- 9.

Bridges in Connect

Connect offers multiple bridging routes:

- Token bridge - wrapped asset bridging
- USDC Bridge - USDC gaslessly and quickly bridged between any CCTP supported chains
- Portal - WH wrapped asset bridging
- - Users can bridge any token between WH supported chains via the TokenBridge contracts that power Portal bridge today
- *
- NEW
- CCTP Based Liquidity Bridge - New price-efficient, fast transfer solution built atop CCTP x WH Messaging
- NEW
- Uni V3 Based Liquidity Bridge - New solution built atop seeded liquidity for WH wrapped assets against native assets in Uni V3 pools
-

Getting Started

It's very easy to add Wormhole Connect to an existing React app.

First, install the npm package.

...

Copy `npm i @wormhole-foundation/wormhole-connect`

...

Now you can use the React component:

...

```
Copy importWormholeConnectfrom'@wormhole-foundation/wormhole-connect';
```

```
functionApp() { return(); }
```

...

Alternative: hosted version via CDN (for any website)

If you're not using React, you can still embed Connect on your website by using the hosted version. Simply copy and paste the following code into your HTML body:

...

Copy

```
scripttype="module"src="https://www.unpkg.com/@wormhole-foundation/[email protected]/dist/main.js"defer
linkrel="https://www.unpkg.com/@wormhole-foundation/[email protected]/dist/main.css"/
```

...

Configuration

The default configuration of Wormhole Connect may not be what you want to use. You may want to provide custom styles or restrict the chains that you allow in your app.

One important set of configuration parameters you should consider changing are the RPC URLs. By default public RPCs are used but they're heavily throttled, so for best user experience, these should be set to custom URLs.

More details on configuration options available is[here](#)

React HTML Tags Configure the Wormhole Connect React component by passing aWormholeConnectConfig object as theconfig attribute

...

```
Copy importWormholeConnect,{ WormholeConnectConfig }from'@wormhole-foundation/wormhole-connect';
```

```
constconfig:WormholeConnectConfig={ networks:["ethereum","polygon","solana"], tokens:
["ETH","WETH","MATIC","WMATIC"], rpcs:{ ethereum:"https://rpc.ankr.com/eth", solana:"https://rpc.ankr.com/solana", } }
```

```
// ...
```

''' If using the hosted version, provideconfig andtheme as JSON-serialized strings on the mount point:

...

Copy

...

Feature Support Matrix

Network	Native Asset	Bridge	Portal	Wrapped Asset	Bridge 0	Slippage	CCTP	USDC Bridge	Gas Dropoff	Gasless																							
Transactions	Ethereum	11/30	✓	✓	✓	✓	✓	Arbitrum	11/30	✓	✓	✓	✓	(USDC Bridge)	Optimism	11/30	✓	✓	✓	✓	(USDC Bridge)												
Avalanche	1/30	✓	✓	✓	✓	Base	11/30	✓	✓	✓	✓	Solana	1/30	✓	12/30	✓	✓	BSC	1/30	✓	N	✓	✓	Polygon	1/30	✓	N	✓	✓				
Fantom	TBD	✓	N	✓	✓	Celo	TBD	✓	N	✓	✓	Moonbeam	TBD	✓	N	✓	✓	Sui	TBD	✓	N	✓	✓	Aptos	TBD	✓	N	N	N	Sei	TBD	✓	N
N N	Osmosis	TBD	✓	N	N	Y (only destination)	Evmos	TBD	✓	N	N	Y (only destination)	Kujira	TBD	✓	N	N	Y (only destination)	CosmosHub	TBD	✓	N	N	Y (only destination)									

Last updated20 hours ago

On this page * [Summary](#) * [Demo](#) * [Functionality in Connect](#) * [Bridges in Connect](#) * [Getting Started](#) * [Configuration](#) * [Feature Support Matrix](#)

Was this helpful? [Edit on GitHub](#)