UX Best Practices

How to get the most out of Relay and cross chain execution

“Bridges?! Where we’re going, we don’t need bridges!!”Dr. Emmett Brown

Relay is designed to be so fast and cheap that transacting cross-chain is indistinguishable from making regular transactions. This makes it possible to build delightful experiences that reduce friction for users.

Bridging vs Cross Chain Execution

Your app is on Chain A. Should you encourage users to explicitly “bridge” enough funds to that chain, or have them execute actions cross chain “just in time”?

This decision depends on a number of factors:

- Repetition
-
    - If the user needs to take multiple actions, bridging might make sense, but if it’s a single core action (e.g. an NFT mint), cross chain execution is the obvious choice
- Supported Chains
-
    - If the user needs to transact on multiple chains, they are better off executing on the fly, to avoid needing to maintain a balance on every chain

Of course, you can always do both. We recommend making cross-chain execution the default, to minimize the friction for the user to have their first “aha moment”, and then later suggesting that they bridge if they are frequently transacting on a particular chain

Internal vs External

This is a no brainer. Relay makes it incredibly easy to integrate a great bridging experience directly into your app. If you’re linking users out to external bridge, you’re adding unnecessary friction.

Default Chain Selection

You want the user to execute a transaction on Chain A, but they’re currently connected to Chain B. Should you have them switch, or just execute cross-chain?

We’ve seen teams take a couple of different approaches:

- Avoid switching completely
-
    - When Boost implemented theirMint Pass
- , they kept things as simple as possible, and just had you mint from whatever chain you were currently connected to. Yes, this meant some users paid more gas than they needed to, but it also reduced friction to the absolute minimum, resulting in over 30,000 mints.
- Only switch for L1
-
    - On Zora, they do something similar to above, except that if the user is currently connected to Ethereum Mainnet, they will suggest switching chains, due to the increased cost of transacting from L1. But if the user is connected to an L2, they will avoid switching chains, even if the user has a balance on the NFT chain.
- Switch if user has balance
-
    - Lastly, you could check if the user has a balance on the chain that you want the user to transact on, and prompt them to switch if they do. This can reduce the gas cost for the user (avoiding the small cross chain overhead), but adds some friction.

Chain Switching UI

Again, we’ve seen different teams take different approaches:

- No UI
-
    - Just use the users’ currently selected chain in their wallet. Simplest and most minimal.
- Chain selector
-
    - Allow the user to choose the chain they want to pay from. What’s nice about this is that you can allow the user to quickly check their balance from multiple chains, to help make their decision. Contract Compatibility Just-In-Time Gas twitter Powered by Mintlify

- Bridging vs Cross Chain Execution

- Internal vs External
- Default Chain Selection
- Chain Switching UI