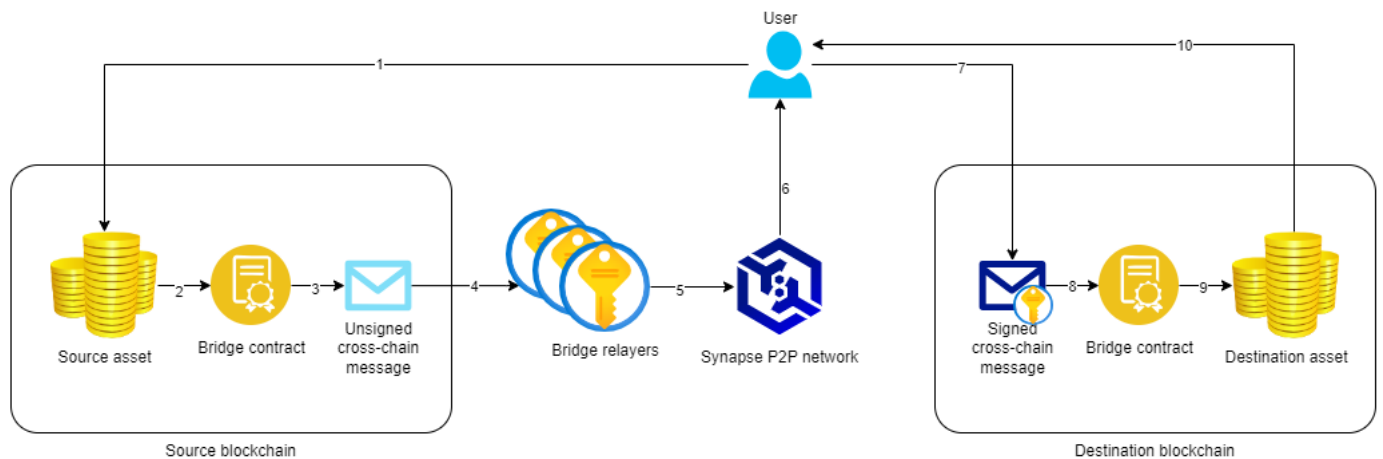# Transaction flow

Each bridge transaction proceeds according to the following scheme:

1. The user initiates a source asset transfer to the bridge contract running on the source blockchain. If the source asset is a native blockchain token, the user transacts to the bridge contract's `transfer()` method. If the source asset is an ERC-20 token, the user transacts to the token's `approve()` method, then transacts to the bridge contract's `transferERC20()` method.
2. If the user sent the original token to get wrapped one, the token is locked in the bridge contract. If the user sent the wrapped token to recover the original asset, the wrapped token is burned.
3. The bridge contract generates a cross-chain message - a bytes string that contains all transaction details. The `MessageCreated()` event is emitted.
4. Bridge relayers constantly listen for bridge contract events. After receiving information about a new message, each relayer performs a checks according to a deterministic algorithm whether it should sign this message or not. Only 8 relayers out of all will be delegated to sign a certain message.
5. Selected relayers sign the message with their private key and send it to the user using the Synapse network.
6. The user waits for all 8 signatures to be received and checks their validity and compliance with the algorithm rules.
7. The user sends a cross-chain message with 8 signatures to the bridge contract running on the destination chain by transacting to the `messageProcess()` method. In the same transaction, the user pays a bridge fee.
8. The bridge contract validates the message, checks whether the signatures are correct and come from exactly those relayers that were selected by the algorithm.
9. If the user unwraps the wrapped tokens, the bridge contract sends the original assets back to the user's wallet. If the user wraps assets, the bridge contract mints the wrapped token to the user's wallet. Additionally the bridge contract divides the received fee into 8 equal amounts and adds them to the balances of relayers involved in a given transaction

processing.

10. The user receives the destination assets on the destination blockchain.