

# Chain Abstraction

Chain abstraction is one of the flagship use cases of Connex. Chain abstraction allows a dApp to execute logic from any chain without requiring users to switch networks, sign transactions on a different chain, and spend gas on a different chain. This pattern can be used at higher layers to fully abstract chains from the user, removing the need for users to consciously have to think about what chain they are on.

For example, a chain abstraction layer for Aave would involve a simple, two-step process.

1. The Connex SDK is used to construct an xcall
2. transaction to be sent by the user on their origin chain.
3. An adapter contract deployed on the destination chain forwards the deposit
4. call to Aave.
- 5.

In the end, no changes are needed on the Aave contracts themselves and cross-chain deposits are enabled for users with maximally simplified UX.

To get started, check out the [Chain Abstraction guide](#).

[Previous Connex and xERC20s Next Chain Abstraction Guide](#) Last updated 8 days ago On this page [Edit on GitHub](#)