



Lightning <> Liquid Swaps

by Boltz

What is Boltz?

Non-Custodial Bitcoin Exchange & Lightning Service provider (LSP)

- Swap $\$$ -BTC/BTC, $\$$ -BTC/L-BTC
- On the fly channel creation
- Differentiation: atomic swaps - non-custodial
- Differentiation: web interface, usable via API

What is Boltz?

Privacy first

- All services, including API exposed via Tor
- No user-identifying data logged or stored

Stability

- Operating since 2019
- One of the oldest & largest lightning nodes

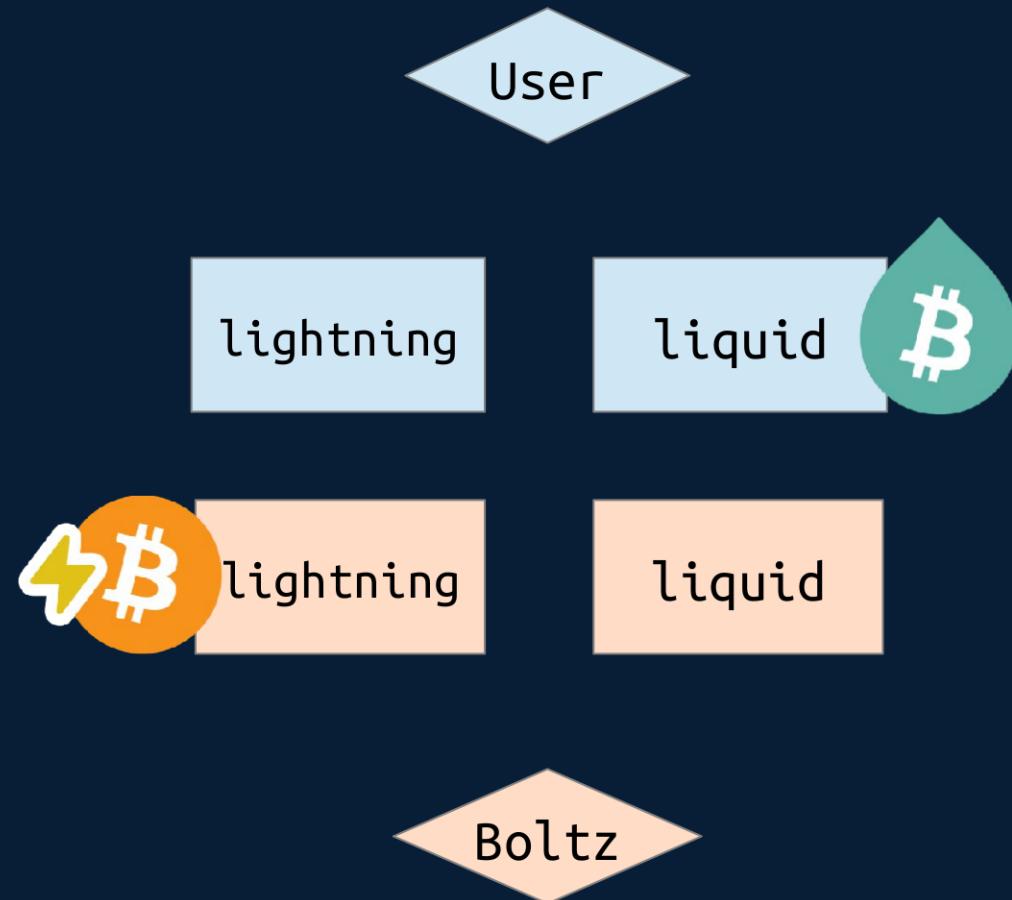
Who uses Boltz?

- Lightning Node operators
 - Rebalance channels (focus of Liquid Swaps)
 - Get a new channel (to be extended)
- Lightning Wallets: receive from/pay to on-chain
- Mainchain Wallets: receive from/pay to lightning
- Liquid Wallets
 - receive from/pay to lightning & mainchain
- Do more with your Bitcoin
 - Convert to L-BTC to use financial products like [fuji.money](#)

What are Atomic Swaps?

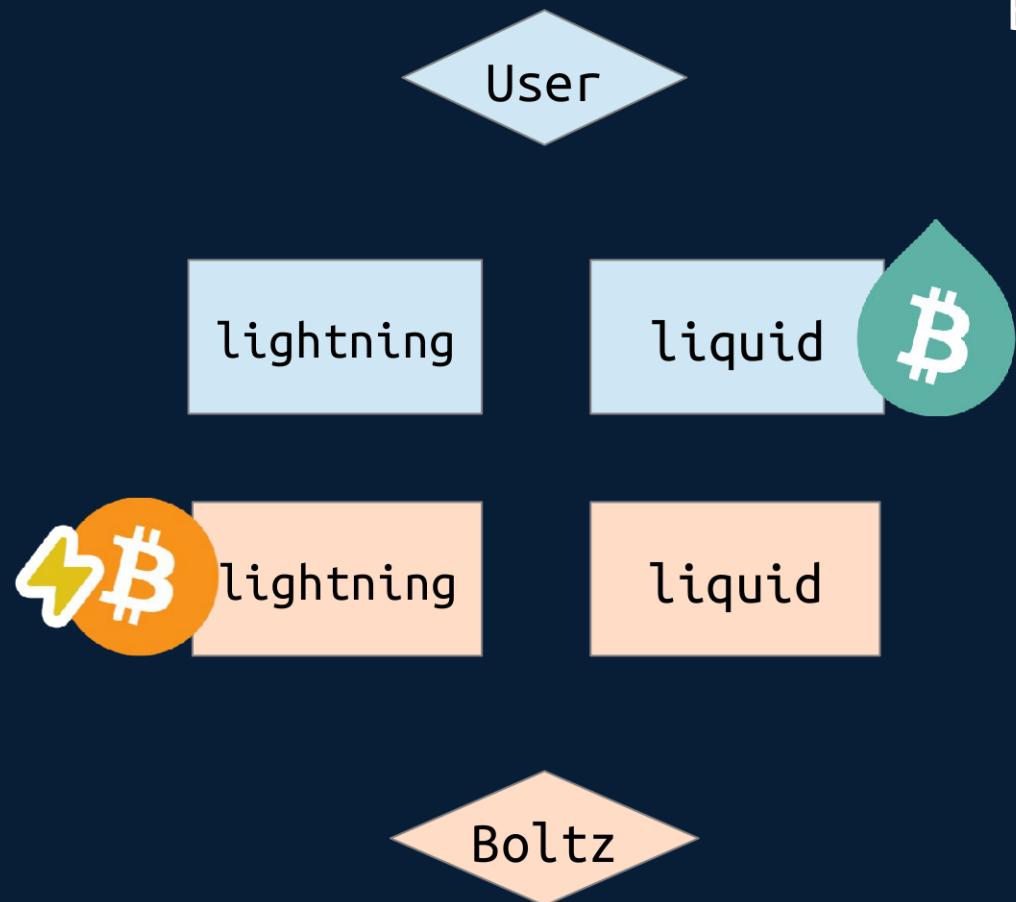
- A way to swap two coins secured by cryptography so that no party can cheat the other.
- The swap only executes if both transactions execute. If 1 out of the 2 does *not* execute, the swap gets refunded.
- In contrast to many definitions found online, they don't have to be between two different chains or P2P.
- Very important for us as a swap provider: this means we are never control customer funds, not even for a split-second

Atomic Swaps - Submarine Swaps

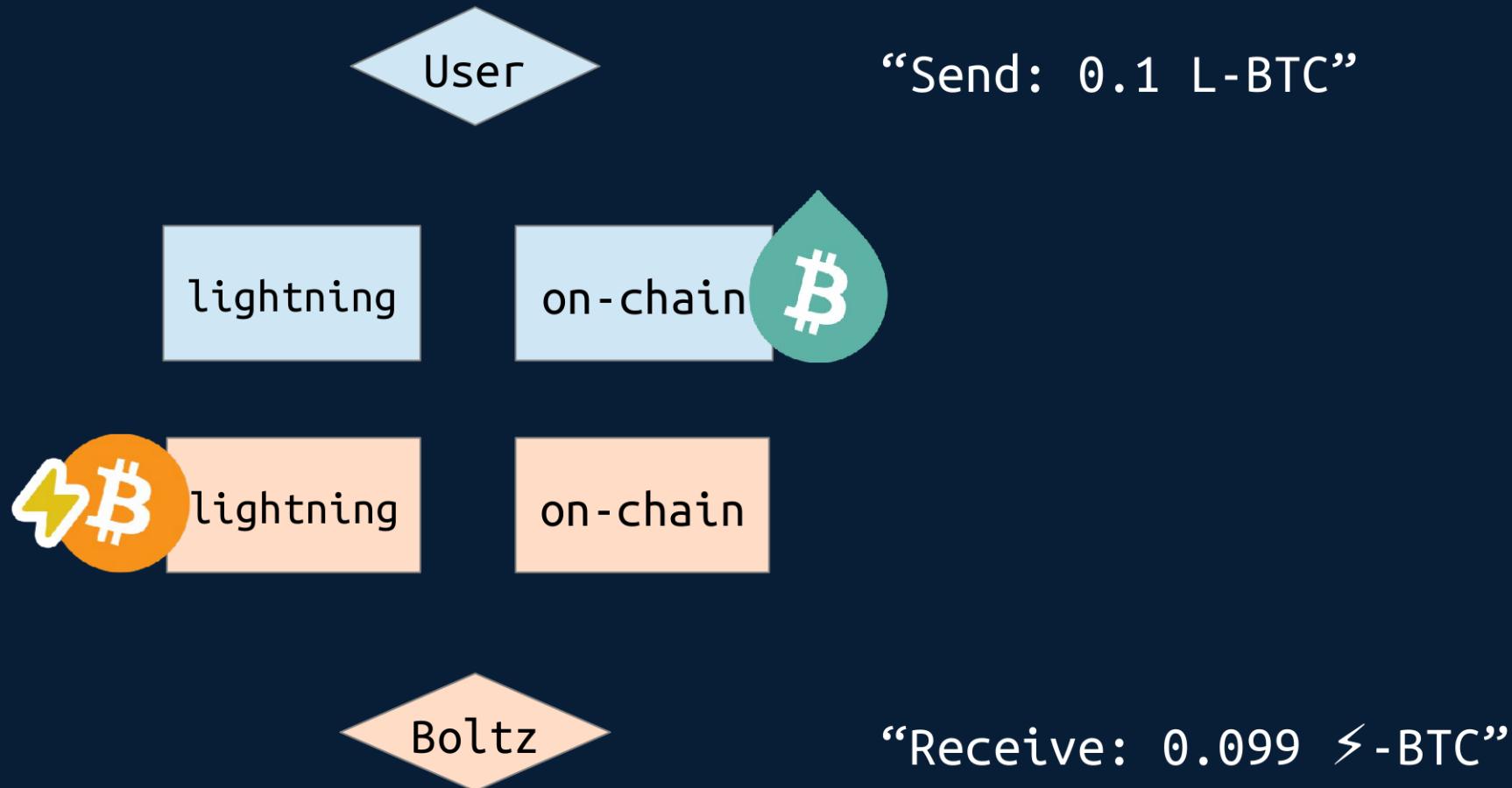


Normal Submarine Swaps (L-BTC -> ₡-BTC)

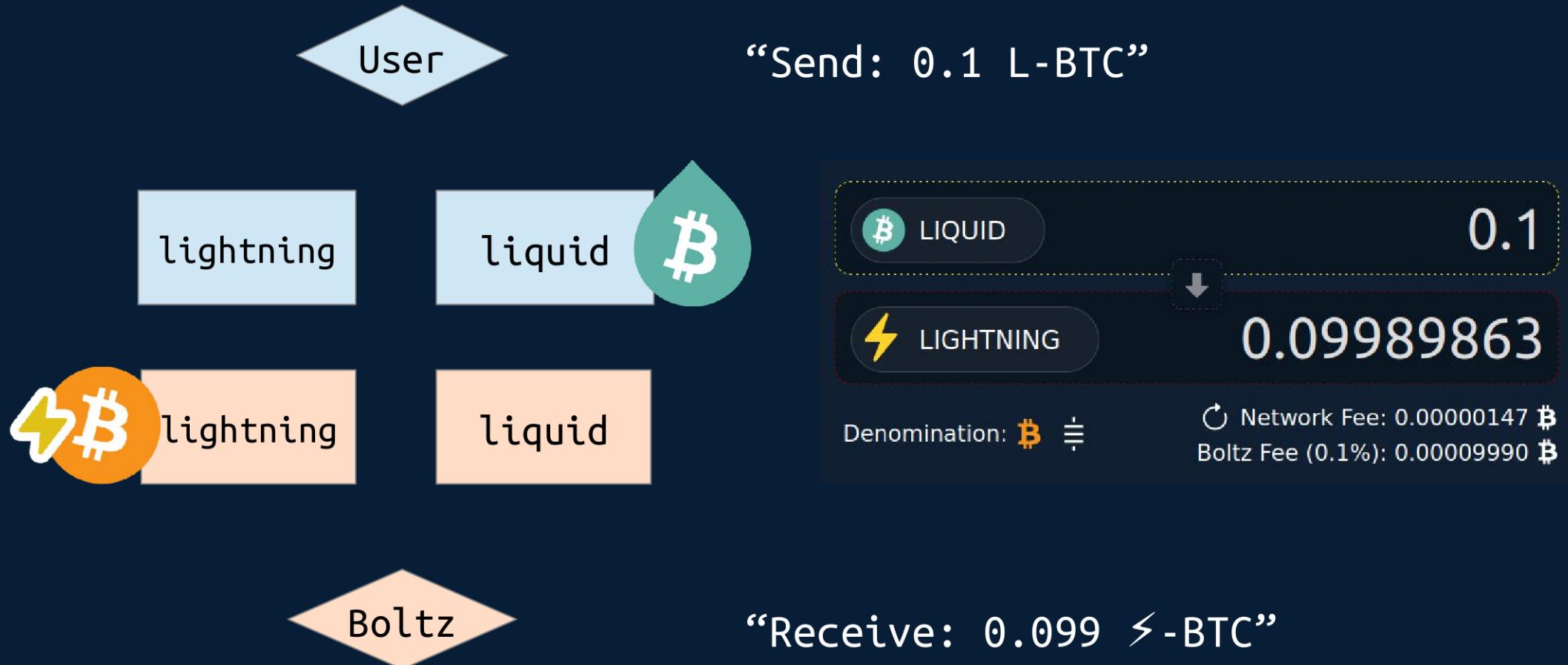
Boltz Swap Protocol (simplified):



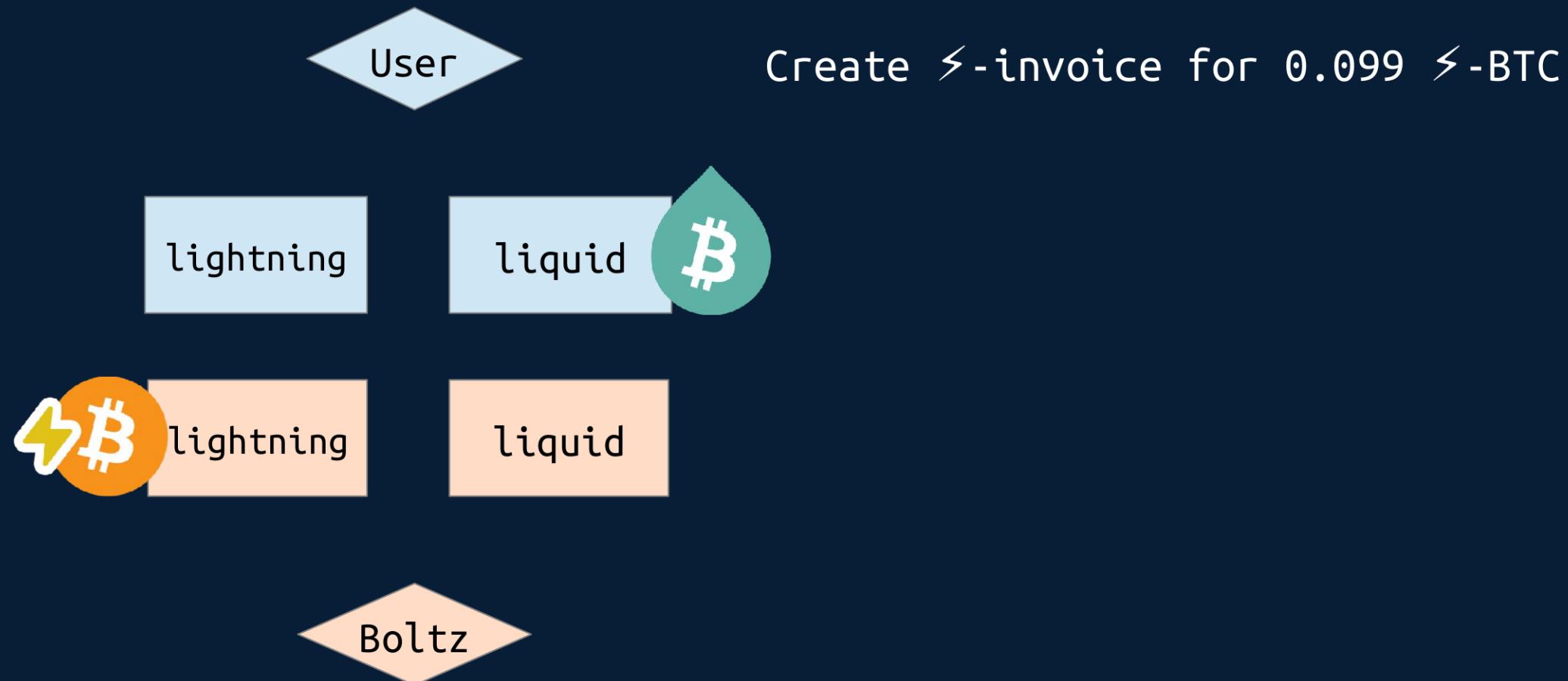
Normal Submarine Swaps (L-BTC -> ₡-BTC)



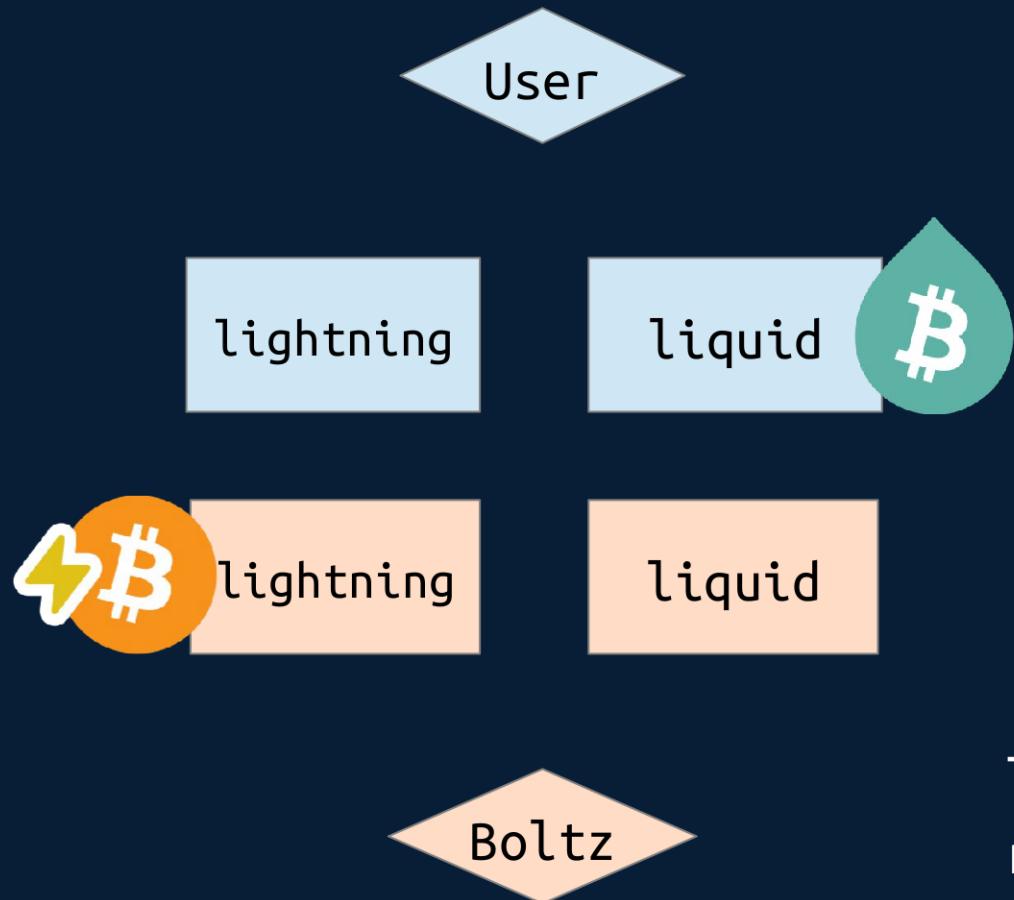
Normal Submarine Swaps (L-BTC -> ₡-BTC)



Normal Submarine Swaps (L-BTC -> ₿-BTC)

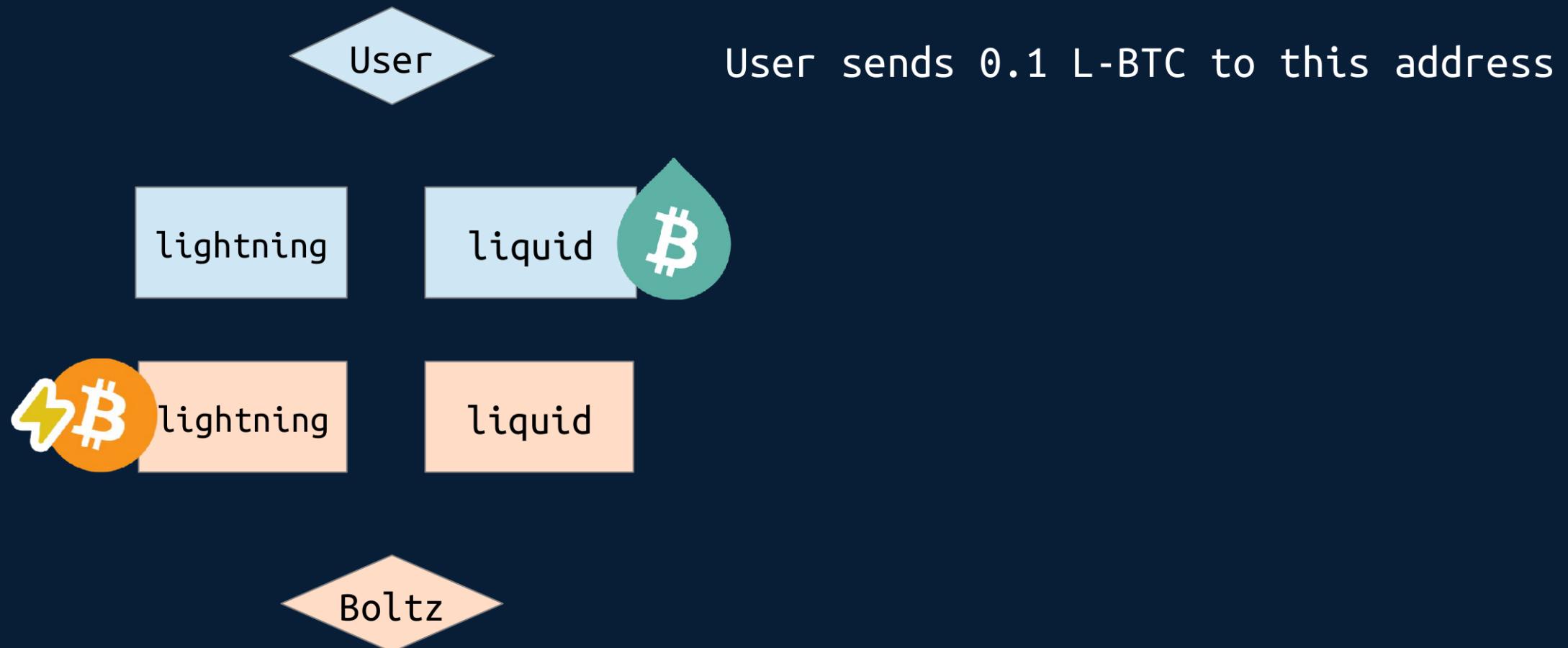


Normal Submarine Swaps (L-BTC -> ₡-BTC)

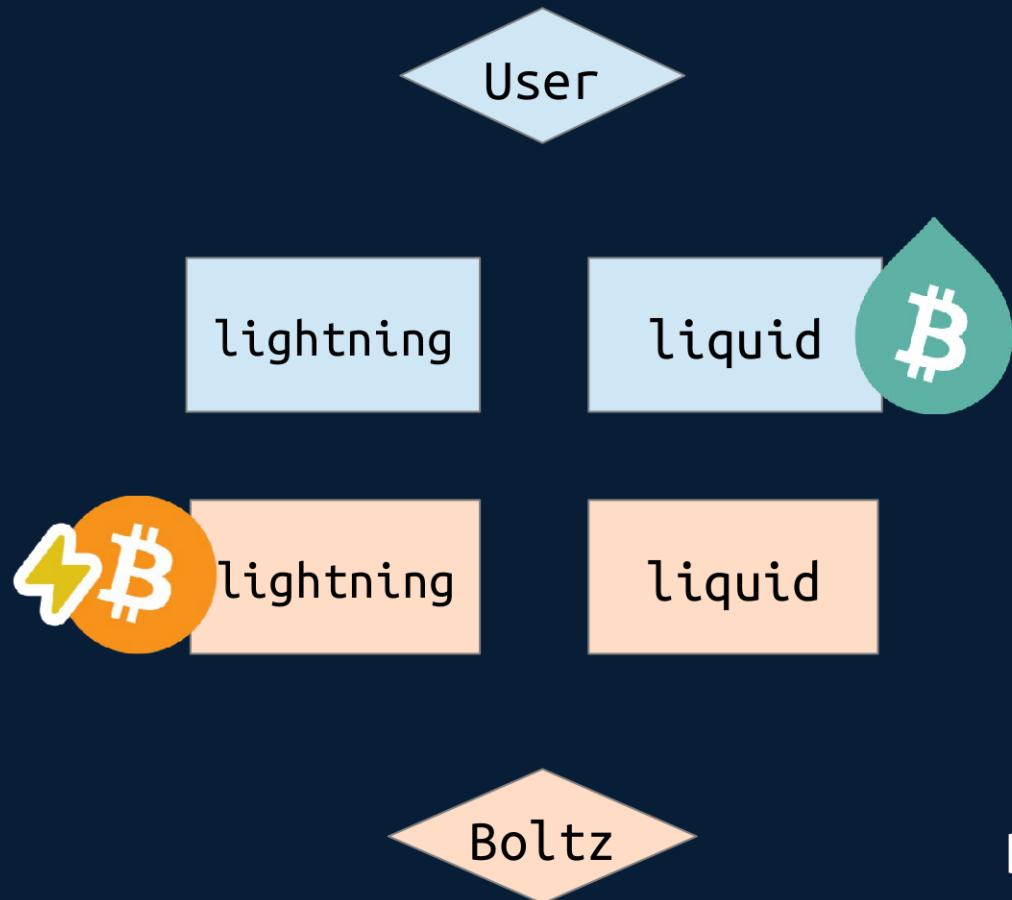


Take invoice preimage hash and create
redeem script to generate liquid address
for user to send 0.1 L-BTC to

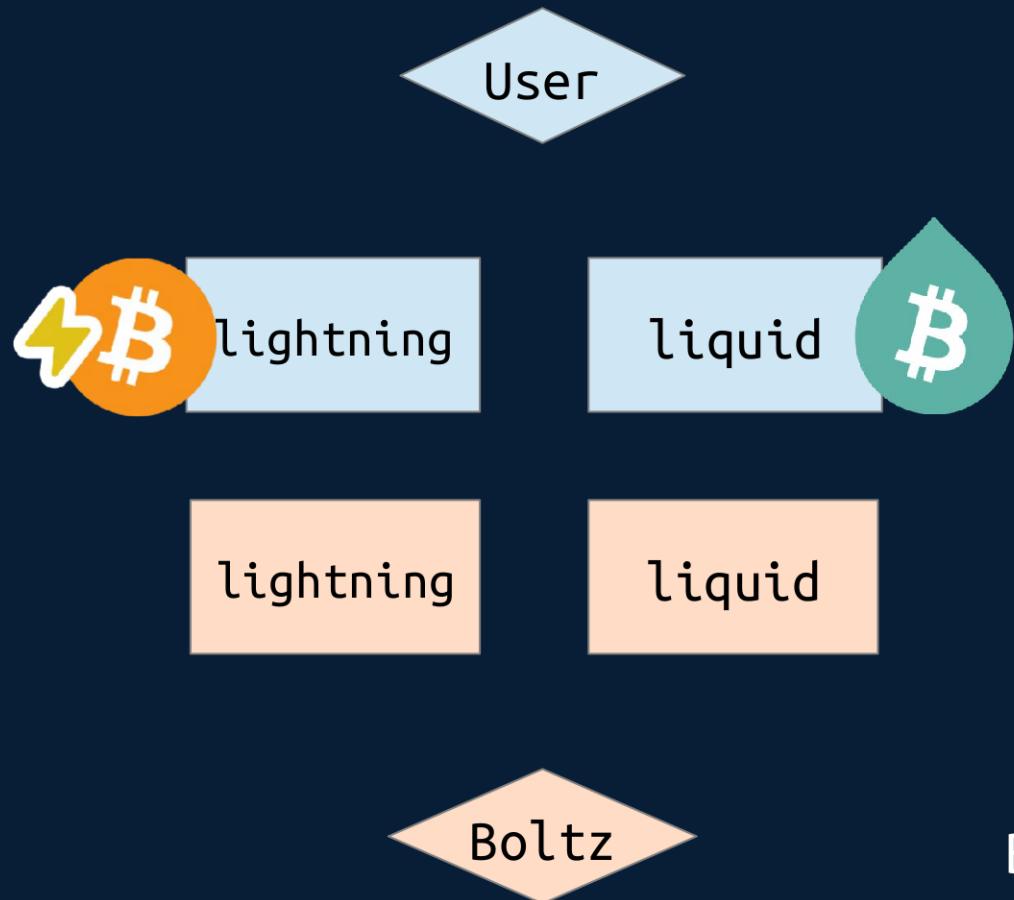
Normal Submarine Swaps (L-BTC -> ₡-BTC)



Normal Submarine Swaps (L-BTC -> ₡-BTC)

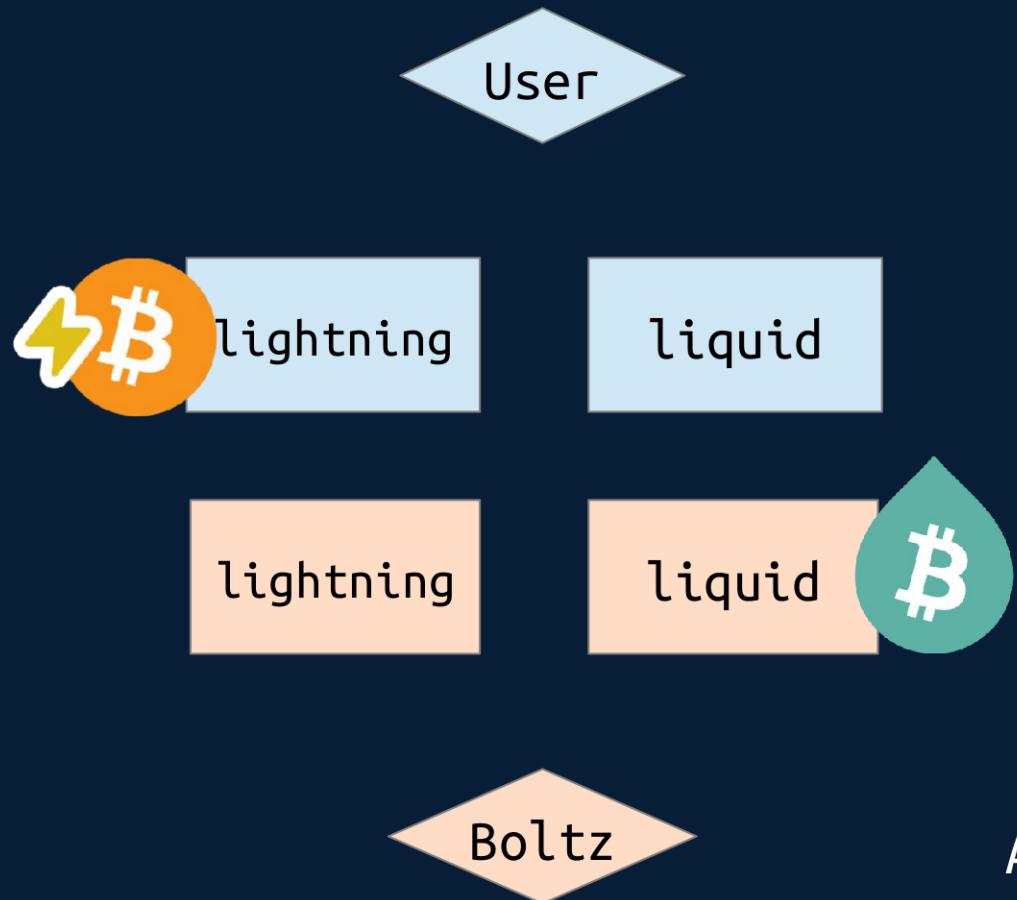


Normal Submarine Swaps (L-BTC -> ₿-BTC)



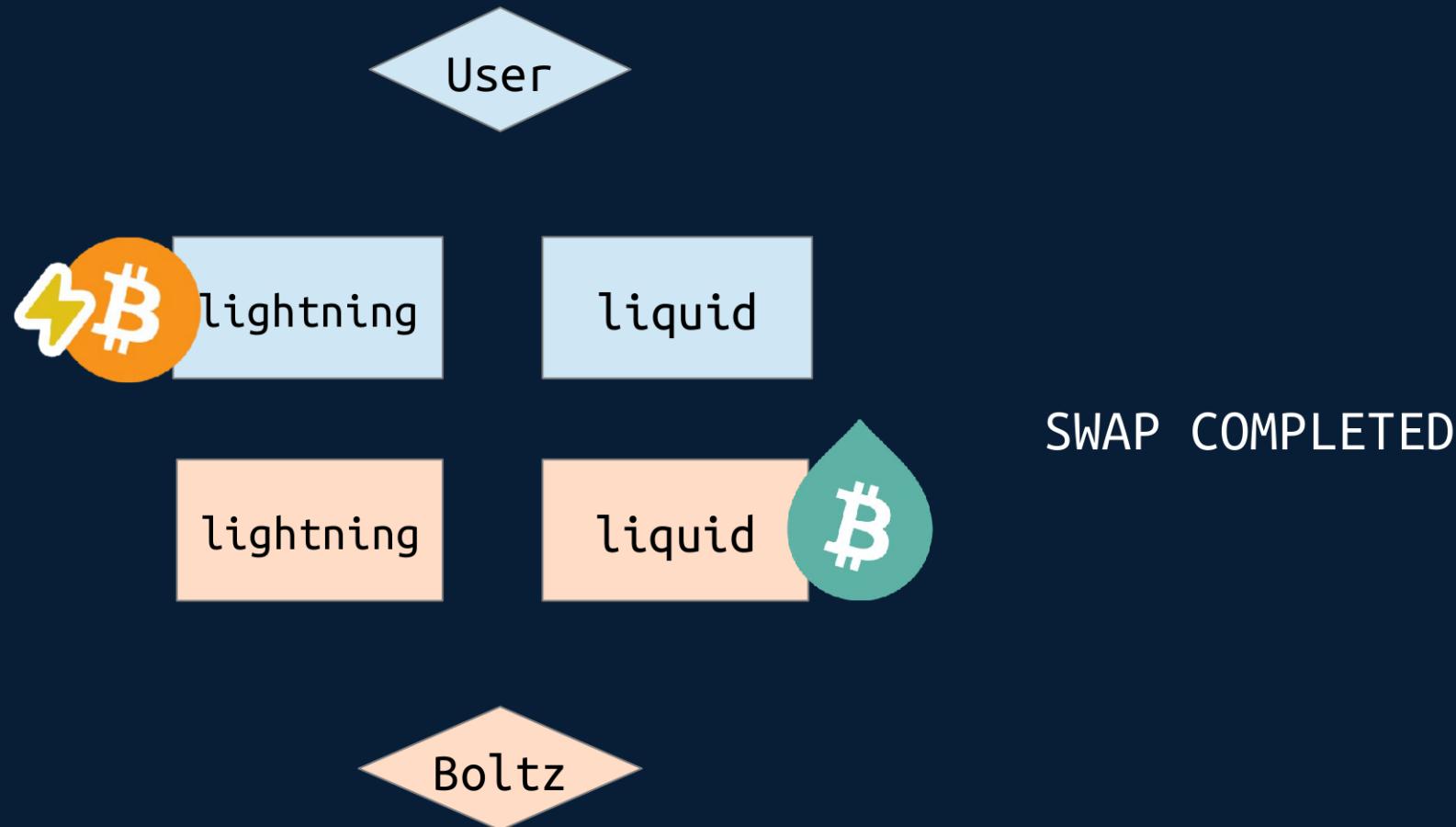
Boltz pays 0.099 lightning invoice,
because of this, preimage gets revealed

Normal Submarine Swaps (L-BTC -> ₡-BTC)



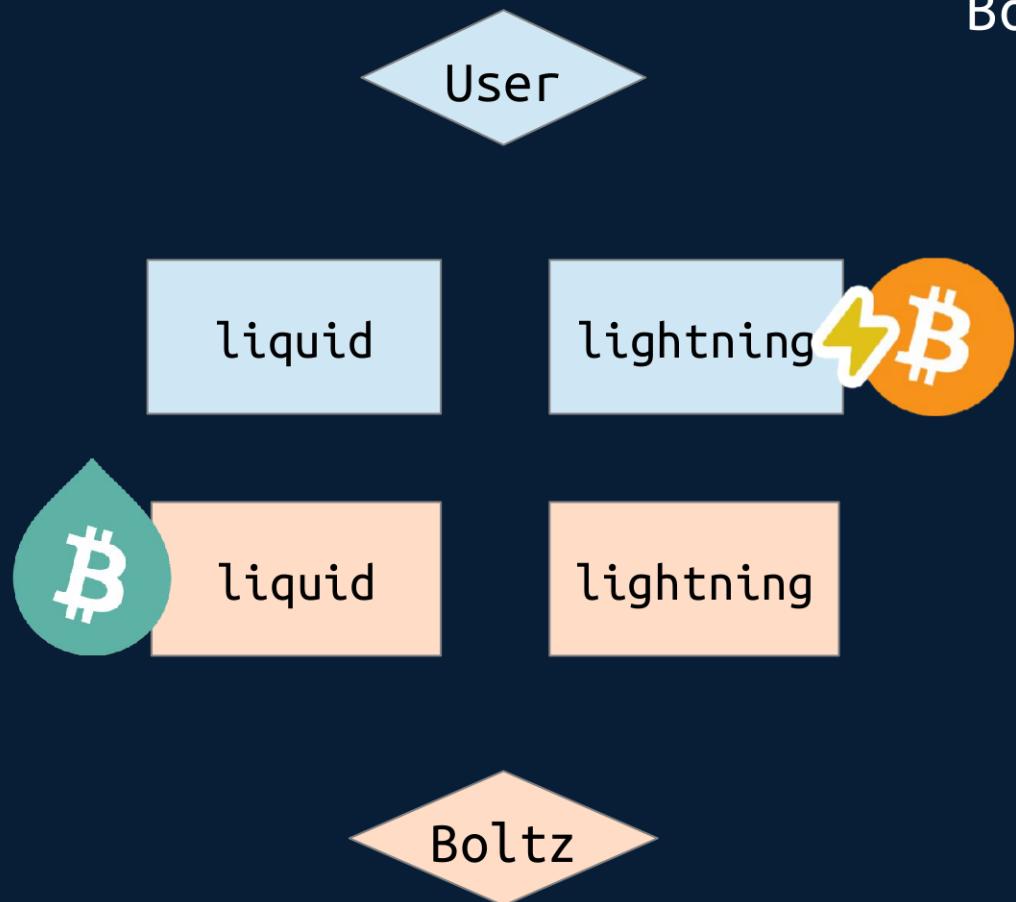
And Boltz can claim 0.1 L-BTC on Liquid
from redeem script

Normal Submarine Swaps (L-BTC → ₡-BTC)

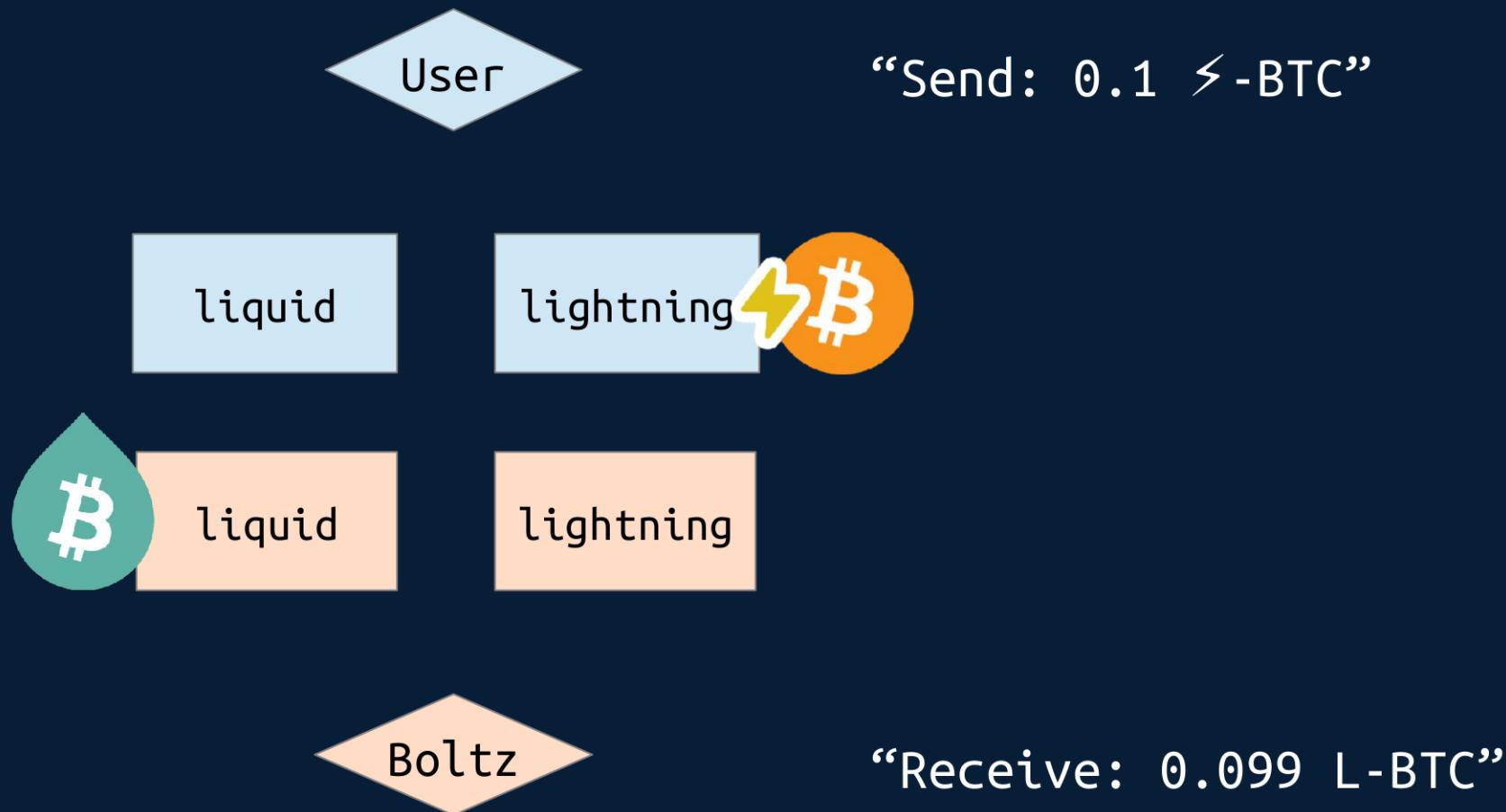


Reverse Submarine Swaps (\leftarrow -BTC -> L-BTC)

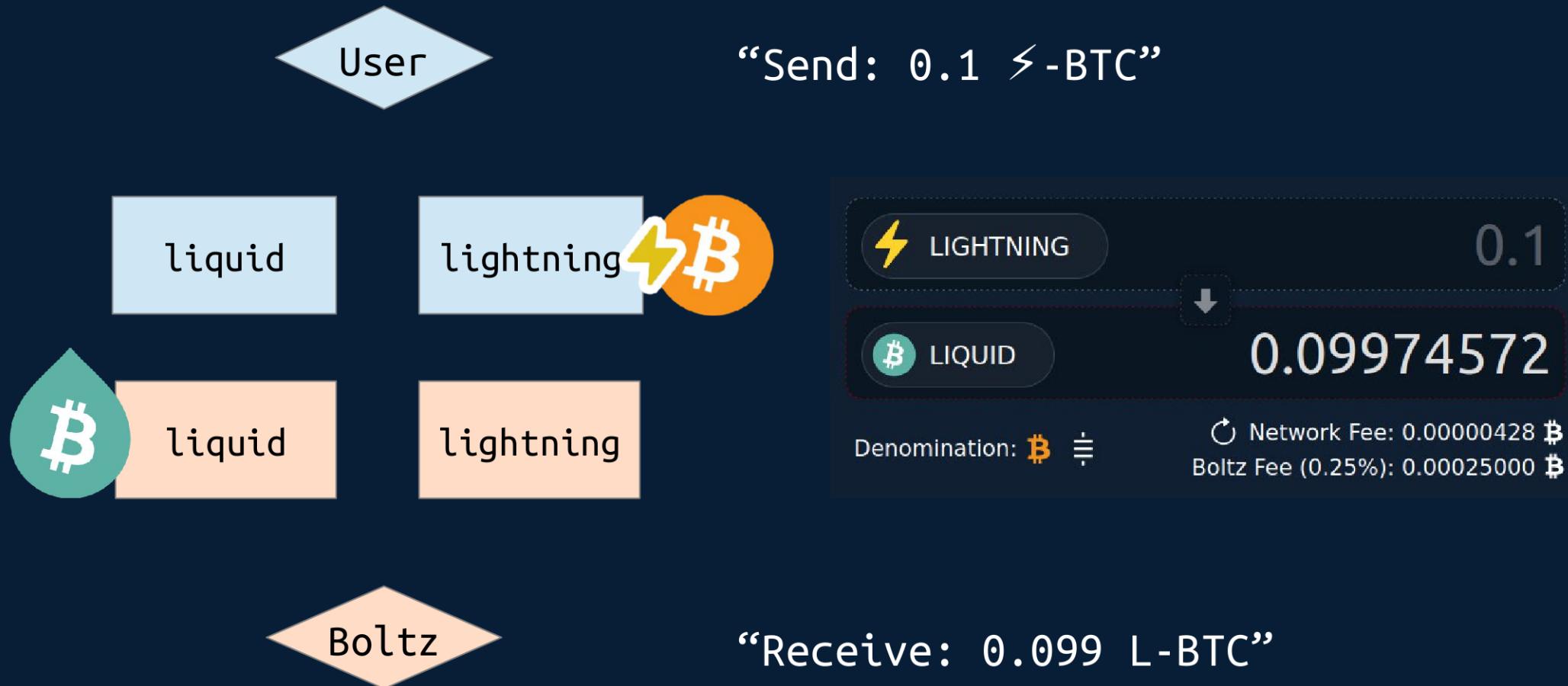
Boltz Swap Protocol (simplified):



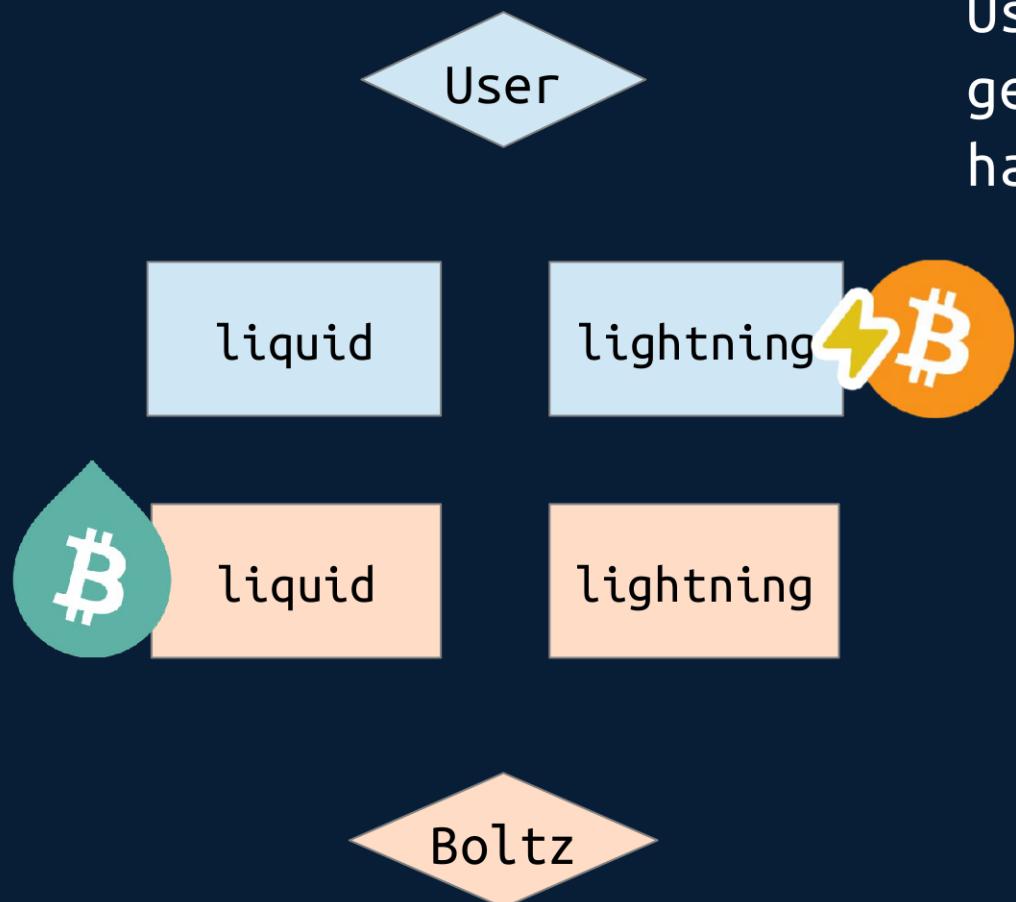
Reverse Submarine Swaps ($\text{S-BTC} \rightarrow \text{L-BTC}$)



Reverse Submarine Swaps ($\text{S-BTC} \rightarrow \text{L-BTC}$)

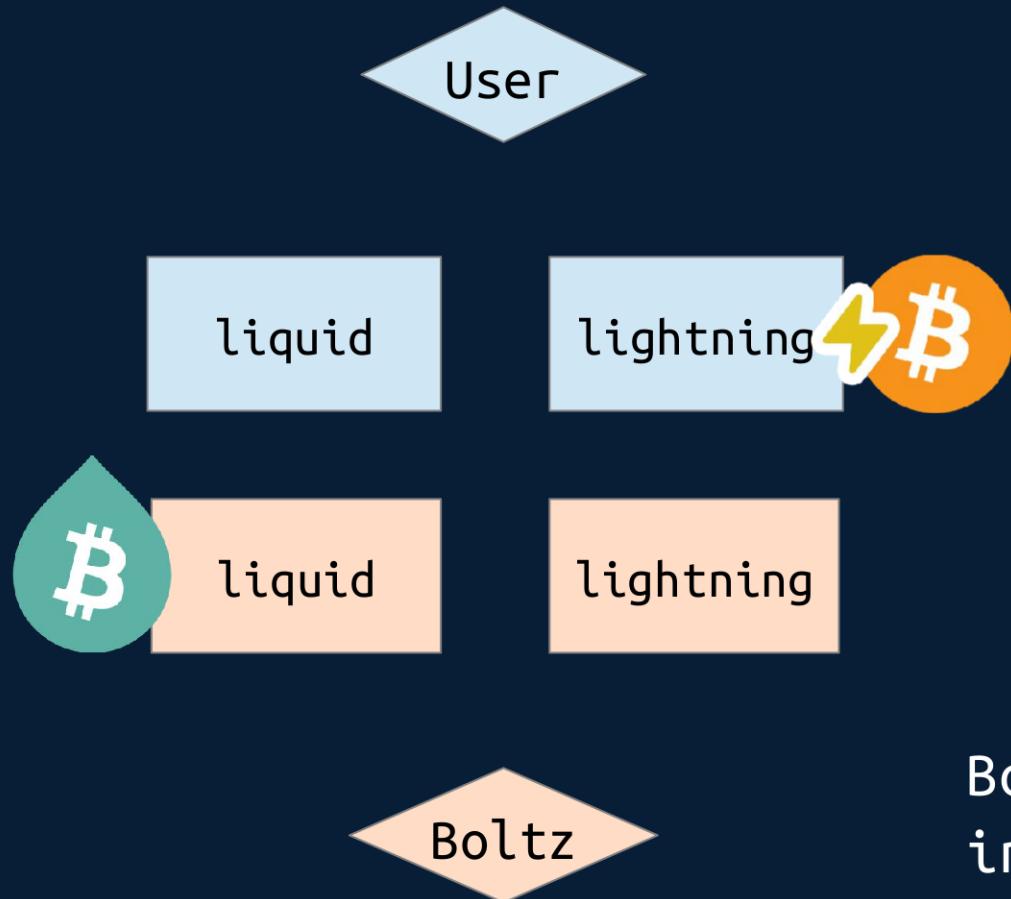


Reverse Submarine Swaps (\leftarrow -BTC -> L-BTC)



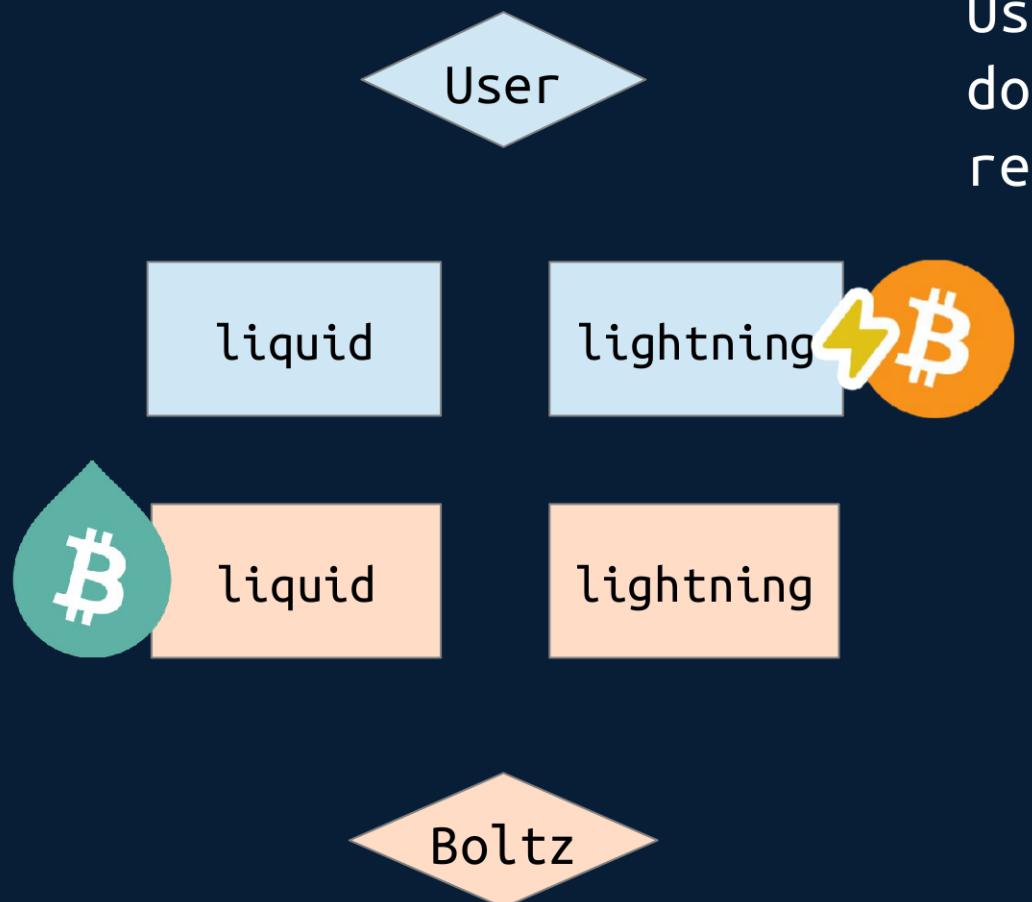
User's app (Liquid.boltz.exchange) generates a preimage, creates SHA256 hash of it and sends hash to Boltz

Reverse Submarine Swaps ($\text{S-BTC} \rightarrow \text{L-BTC}$)



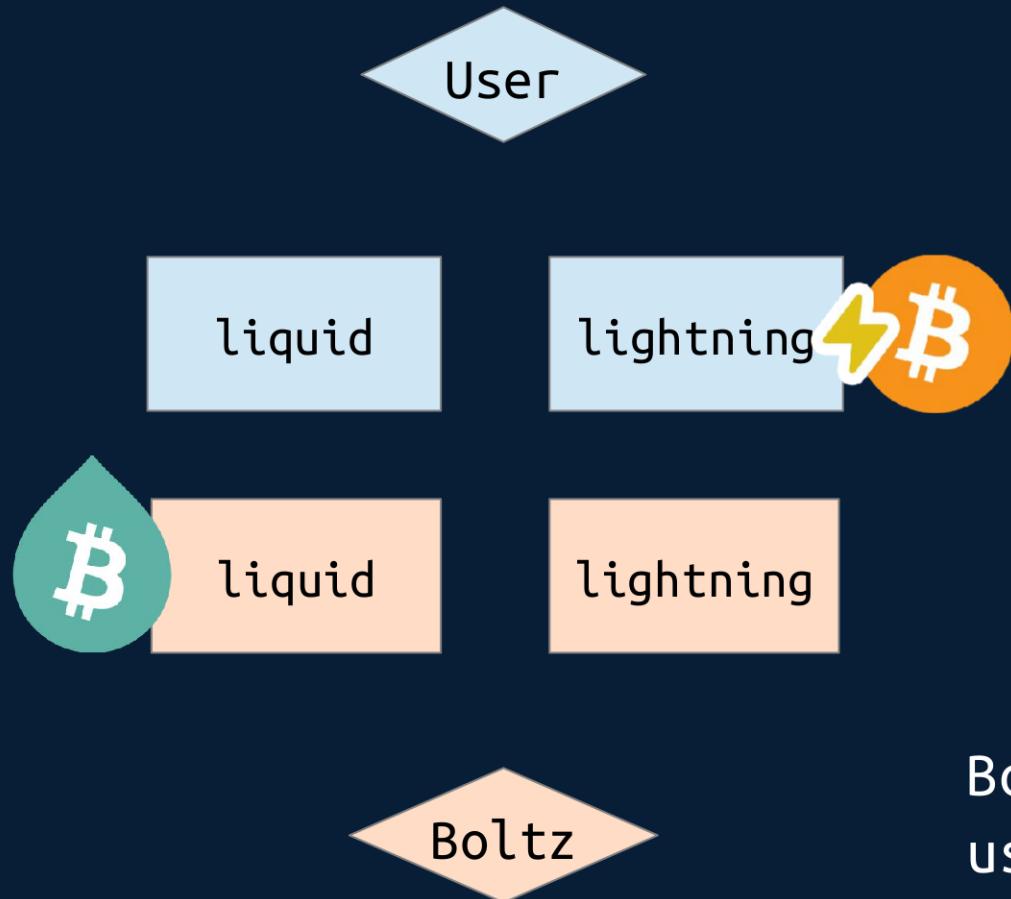
Boltz creates a so-called “hold invoice” about 0.1 S-BTC
with hash received from user

Reverse Submarine Swaps (\leftarrow -BTC -> L-BTC)



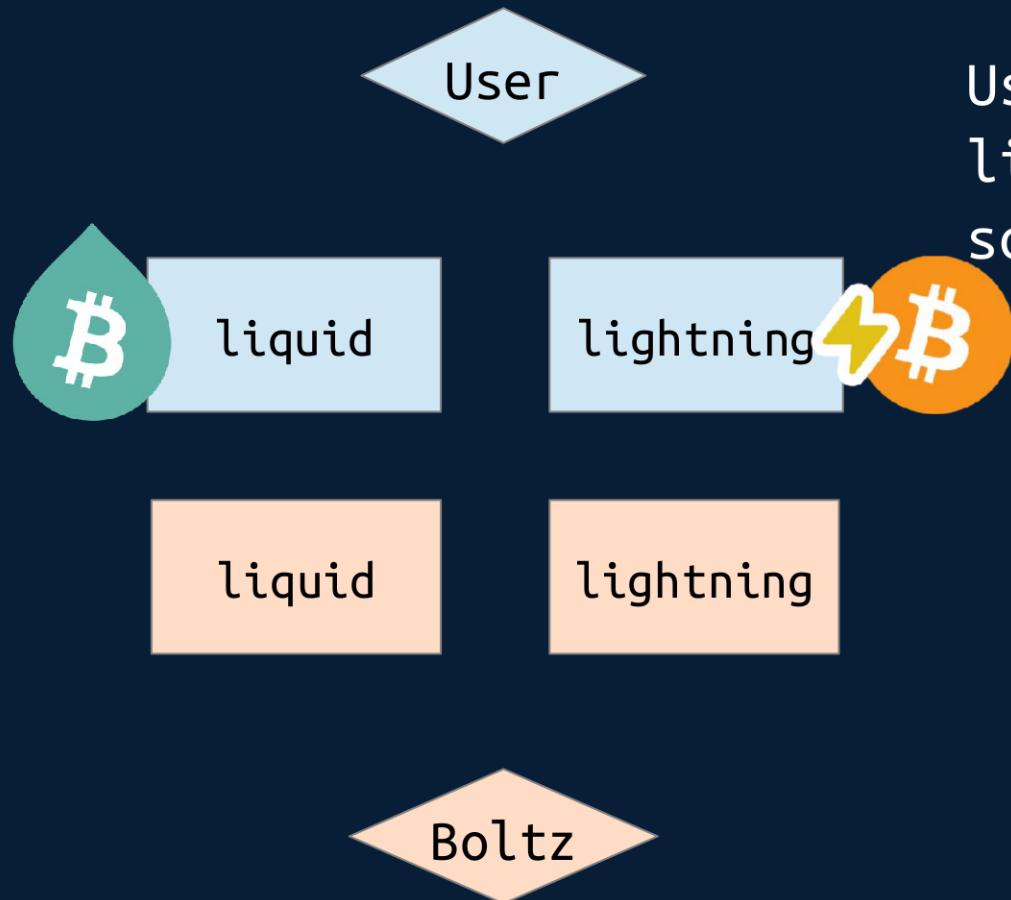
User pays Boltz's invoice, but invoice does not settle because user doesn't reveal preimage to Boltz yet

Reverse Submarine Swaps ($\text{S-BTC} \rightarrow \text{L-BTC}$)

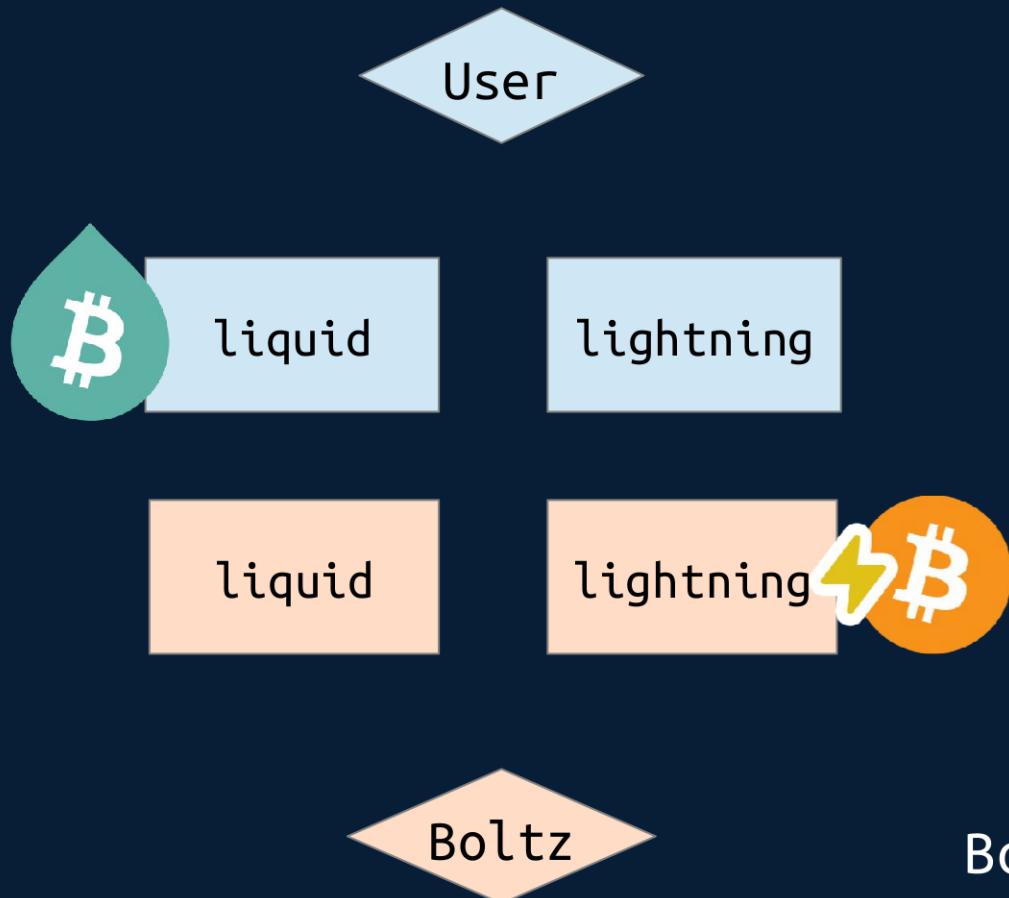


Boltz locks up 0.099 L-BTC on Liquid
using **same hash from user**

Reverse Submarine Swaps ($\$-\text{BTC} \rightarrow \text{L-BTC}$)

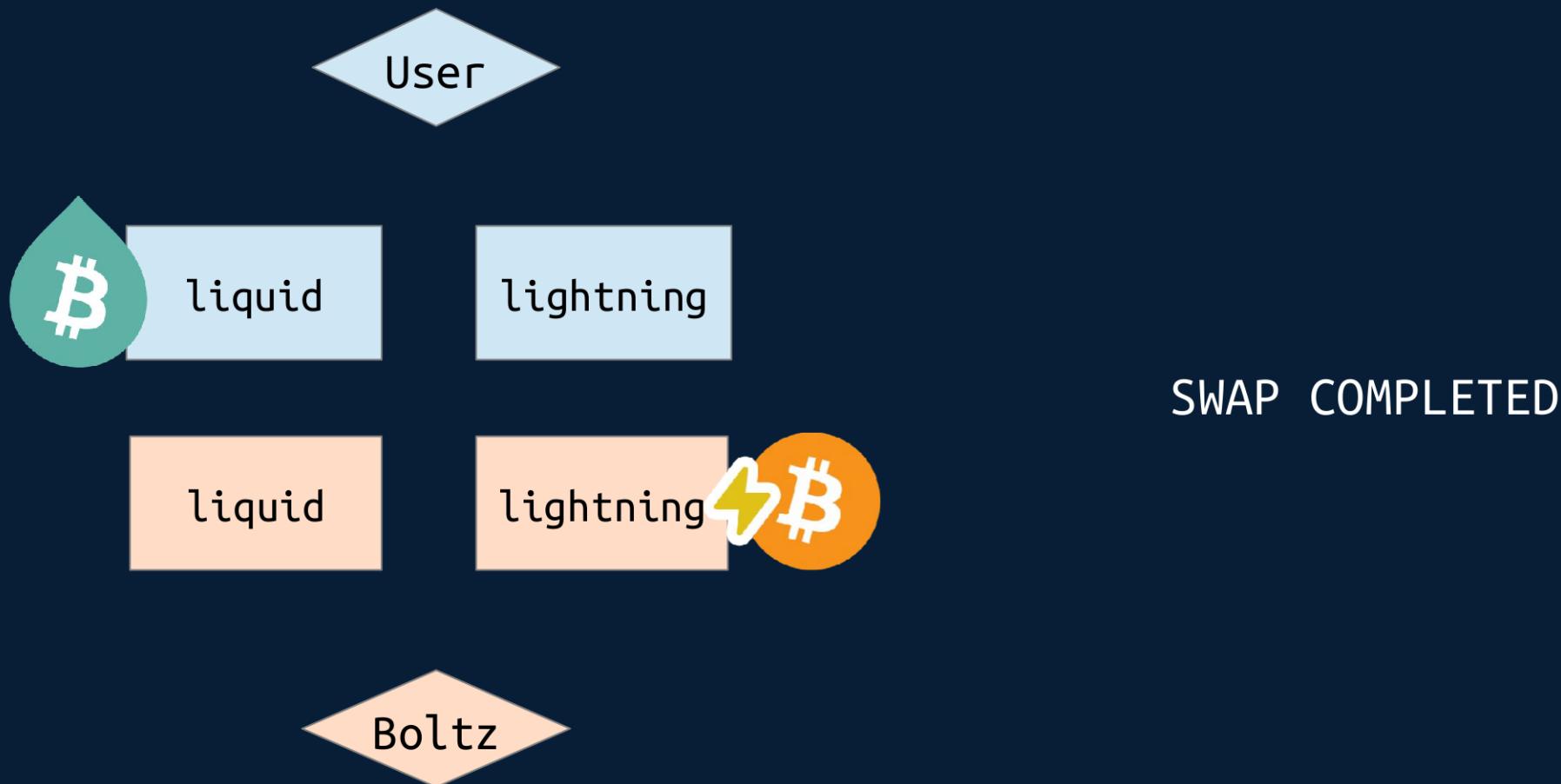


Reverse Submarine Swaps ($\text{S-BTC} \rightarrow \text{L-BTC}$)



Boltz detects preimage in users claim transaction and uses it to settle 0.1 BTC lightning invoice

Reverse Submarine Swaps ($\text{S-BTC} \rightarrow \text{L-BTC}$)



That's it!



The best part

- Reliably cheap (we broadcast with 0.11 sat/vbyte as long as liquid blocks are not full)
- Delinking from mainchain UTXO
- All liquid transactions are confidential

Inputs & Outputs

Details			
 Gxc4NXY8gkxYJVcyshVCBtVq1bPD23xhuh	Confidential	 ex1q545k3qt2d8unsw82q65hdvf... lqnzly07	Confidential 
 H3vappLTFrfVA74JhFx1XSyQbkMkocZgwT	Confidential	Transaction fee	0.00000205 L-BTC 
			Confidential

Resources

liquid.boltz.exchange
docs.boltz.exchange
github.com/BoltzExchange