



Atomic Swaps on ⚡



Atomic Swaps on ⚡

... just hit a snag

What we are building

What we are building

A decentralized exchange protocol.

What we are building

A decentralized exchange protocol.

For centralized exchanges.

What we are building

A decentralized exchange protocol.

For centralized exchanges.

On ⚡lightning⚡.

What we are building

A decentralized exchange protocol.

For centralized exchanges (& individuals).

On ⚡lightning⚡.

What we are building

A decentralized exchange protocol.

For centralized exchanges (Why? Liquidity.)

On ⚡lightning⚡.

Endgame

Every individual who runs XUD, the node software, is part of the network and can trade with everyone else

Atomic Swaps - The Concept

- ◆ Trustless exchange of two different assets
 - ◆ Trustless = no middleman/escrow service
- ◆ How: guarantee atomicity
 - ◆ Both sides of the trade happen or none at all
 - ◆ Technology: Hash Time Lock Contracts (HTLCs)



Atomic Swaps - The Concept



Atomic Swaps - The Concept



preimage



Atomic Swaps - The Concept



$H(\text{preimage})$



Atomic Swaps - The Concept

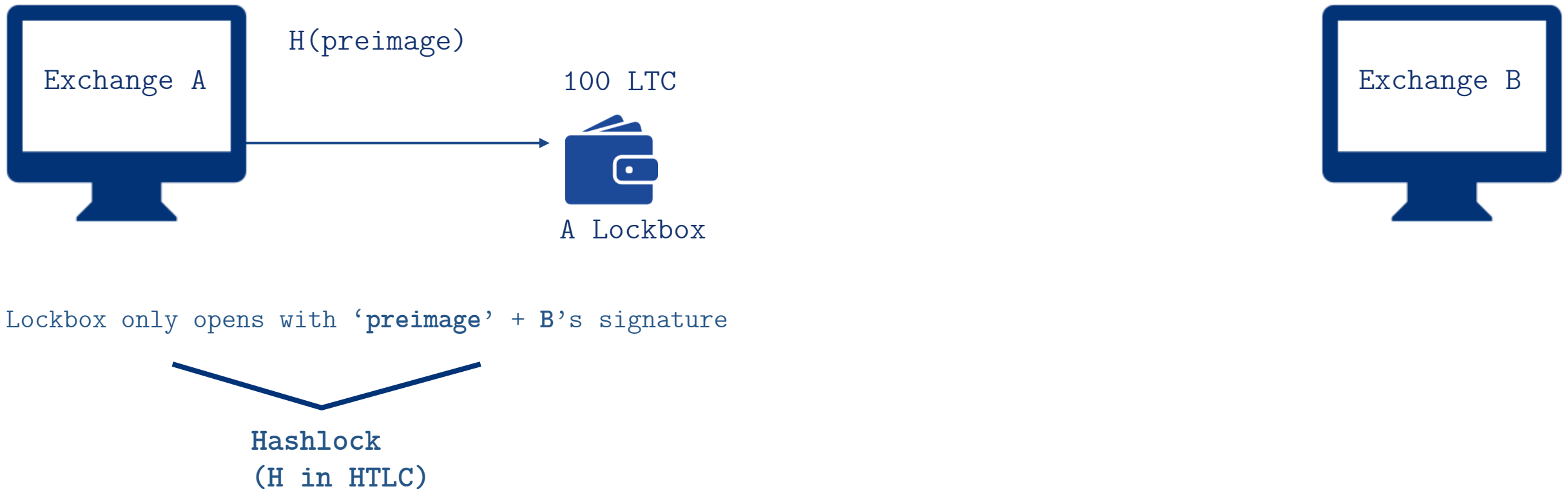


Atomic Swaps - The Concept



Lockbox only opens with 'preimage' + B's signature

Atomic Swaps - The Concept



Atomic Swaps - The Concept



Lockbox only opens with 'preimage' + B's signature

Atomic Swaps - The Concept



Lockbox only opens with 'preimage' + B's signature

Lockbox only opens with 'preimage' + A's signature

Atomic Swaps - The Concept

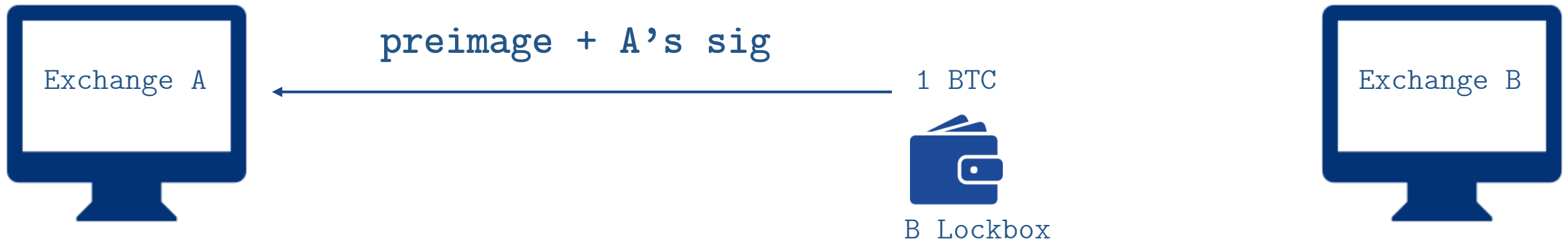


Lockbox only opens with 'preimage' + B's signature

Lockbox only opens with 'preimage' + A's signature

Both lockboxes need the same 'preimage' to be opened!

Atomic Swaps - The Concept



If A wants to open B's lockbox it has to reveal the **preimage** on the blockchain / payment channel

Atomic Swaps - The Concept



preimage



Atomic Swaps - The Concept



Atomic Swaps - The Concept



Atomic Swaps - The Concept



TimeLock (T & L in HTLC):

If something goes wrong, 100 LTC go back to A and 1 BTC back to B after certain time interval (# of blocks)

Atomic Swaps - The Concept



Hash Time Lock Contracts (HTLCs):

On-chain and off-chain. Via BIP199 on Bitcoin and *native* in Lightning payment channels.

Atomic Swaps - The Concept



Hash Time Lock Contracts (HTLCs):

On-chain and off-chain. Via BIP199 on Bitcoin and *native* in Lightning payment channels.

- Scripts on Bitcoin
- Smart Contract on Ethereum

Atomic Swaps - The Concept

OP_IF

OP_SHA256 H(preimage)

OP_EQUALVERIFY OP_DUP OP_HASH160

a_pubkey

OP_ELSE

timeout OP_CHECKSEQUENCEVERIFY OP_DROP

OP_DUP

OP_HASH160 b_pubkey

OP_ENDIF

OP_EQUALVERIFY

OP_CHECKSIG

Source: <https://github.com/bitcoin/bips/blob/master/bip-0199.mediawiki>



@kilrau

Atomic Swaps - The Concept

OP_IF

OP_SHA256 H(preimage)

OP_EQUALVERIFY OP_DUP OP_HASH160

a_pubkey

OP_ELSE

timeout OP_CHECKSEQUENCEVERIFY OP_DROP

OP_DUP

OP_HASH160 b_pubkey

OP_ENDIF

OP_EQUALVERIFY

OP_CHECKSIG

Source: <https://github.com/bitcoin/bips/blob/master/bip-0199.mediawiki>



@kilrau

Atomic Swaps - The Concept

OP_IF

OP_SHA256 H(preimage)

OP_EQUALVERIFY OP_DUP OP_HASH160

a_pubkey

OP_ELSE

timeout OP_CHECKSEQUENCEVERIFY OP_DROP

OP_DUP

OP_HASH160 **b_pubkey**

OP_ENDIF

OP_EQUALVERIFY

OP_CHECKSIG

Source: <https://github.com/bitcoin/bips/blob/master/bip-0199.mediawiki>



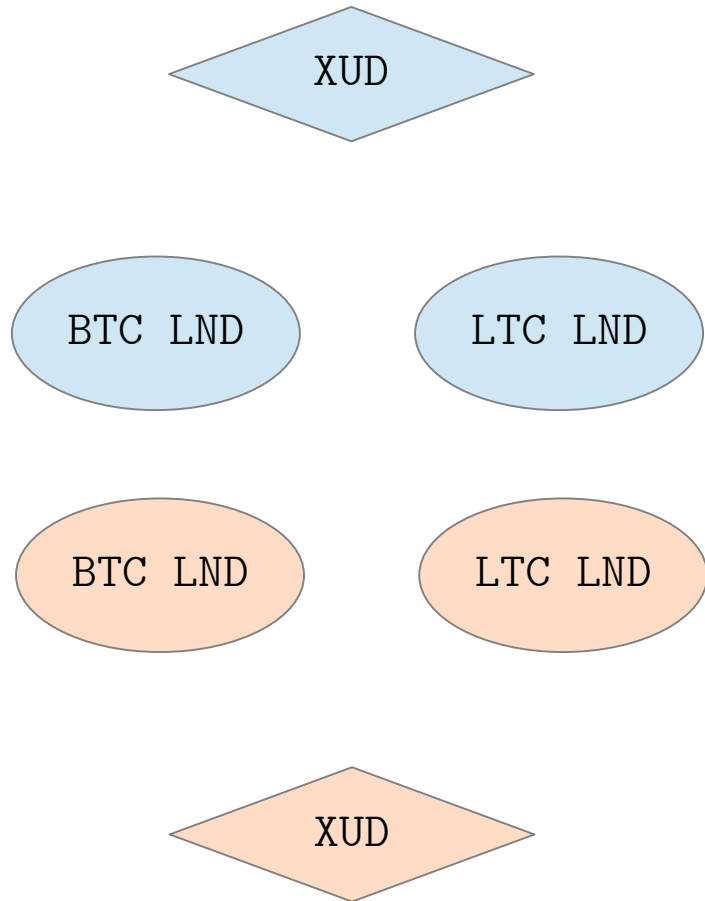
@kilrau

Atomic Swaps on ⚡

Atomic Swaps on ⚡

Goal: minimal changes to
lightning implementation (LND)

Atomic Swaps on



The Setup

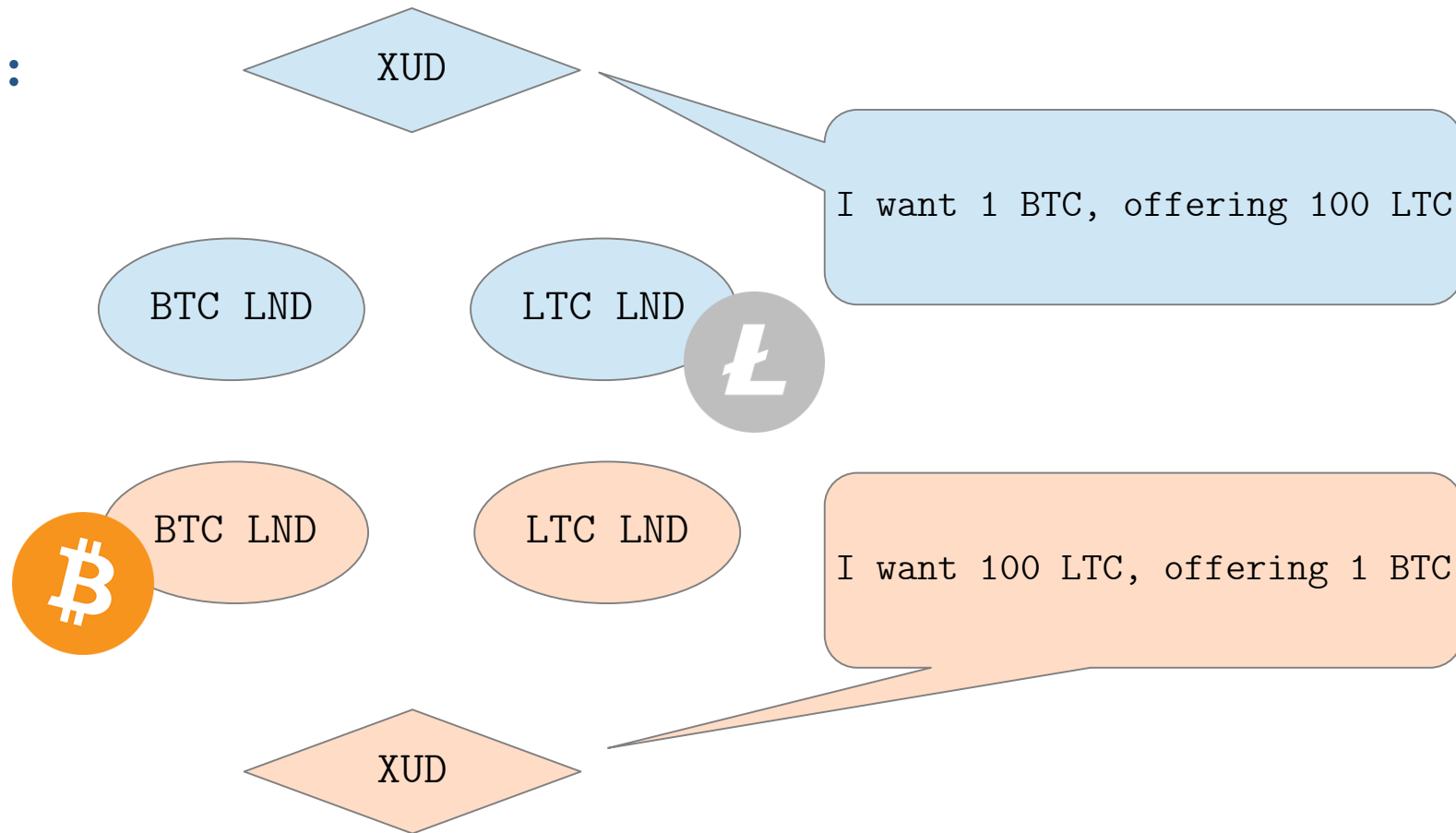
- Two **XUD** that each manage **two LND clients**, one on **BTC** and one on **LTC**.
- LNDs configured to query XUD when they are recipients of a payment with an unknown preimage

Assumptions:

- Routes with sufficient capacity exist between the BTC LND nodes and the LTC LND nodes.
- XUDs can communicate with each other via P2P layer e.g. negotiate a trade, exchange LND pubkeys

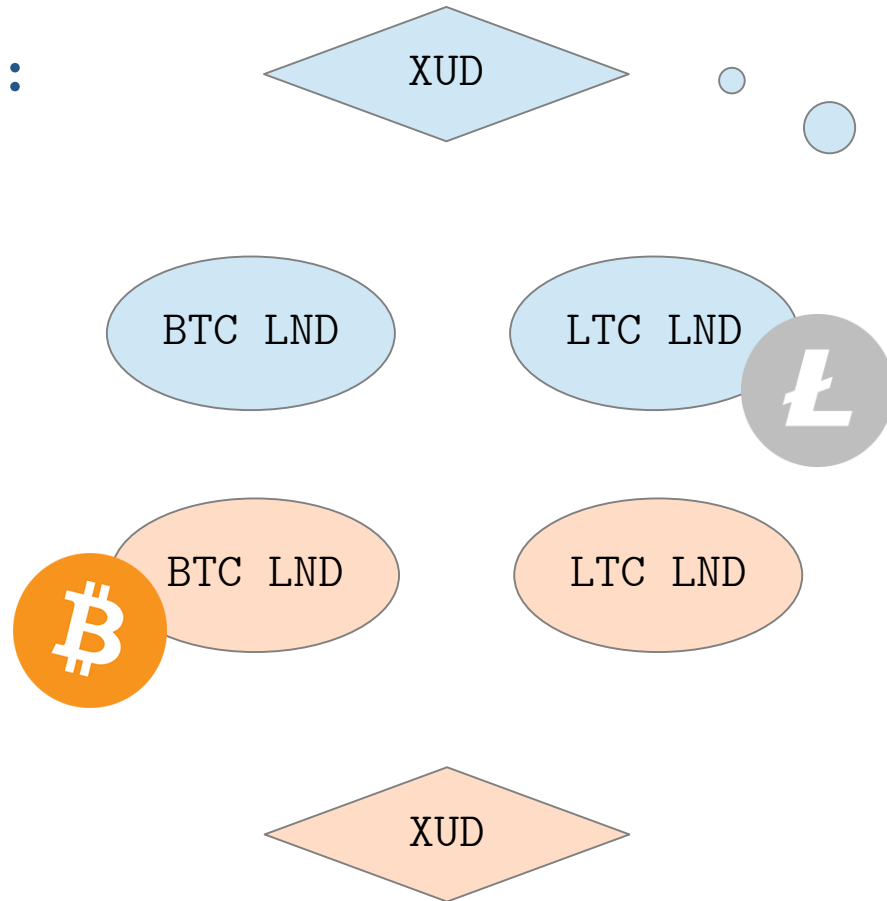
Atomic Swaps on ⚡

Step 0:



Atomic Swaps on ⚡

Step 0:



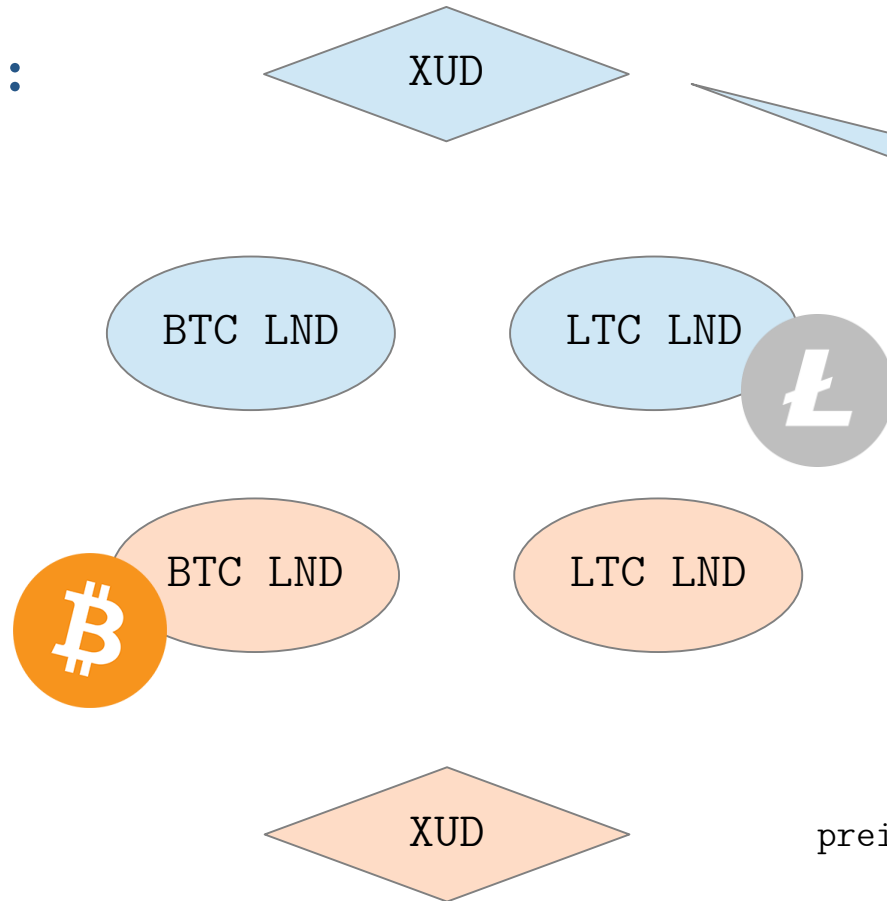
```
preimage = MVHz941lQvHBYRVUE/XSDNbzdrHSNl4f3pANIhin6ZA=
```

```
r_hash =  
3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990
```

I'll initiate the swap by creating a preimage and hashing it...

Atomic Swaps on ⚡

Step 1:



preimage = MVHz9411QvHBYRVUE/XSDNbzdrHSN14f3pANIhin6ZA=

r_hash =

3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

Let's make a swap using
r_hash 3151f3f7896542f1...

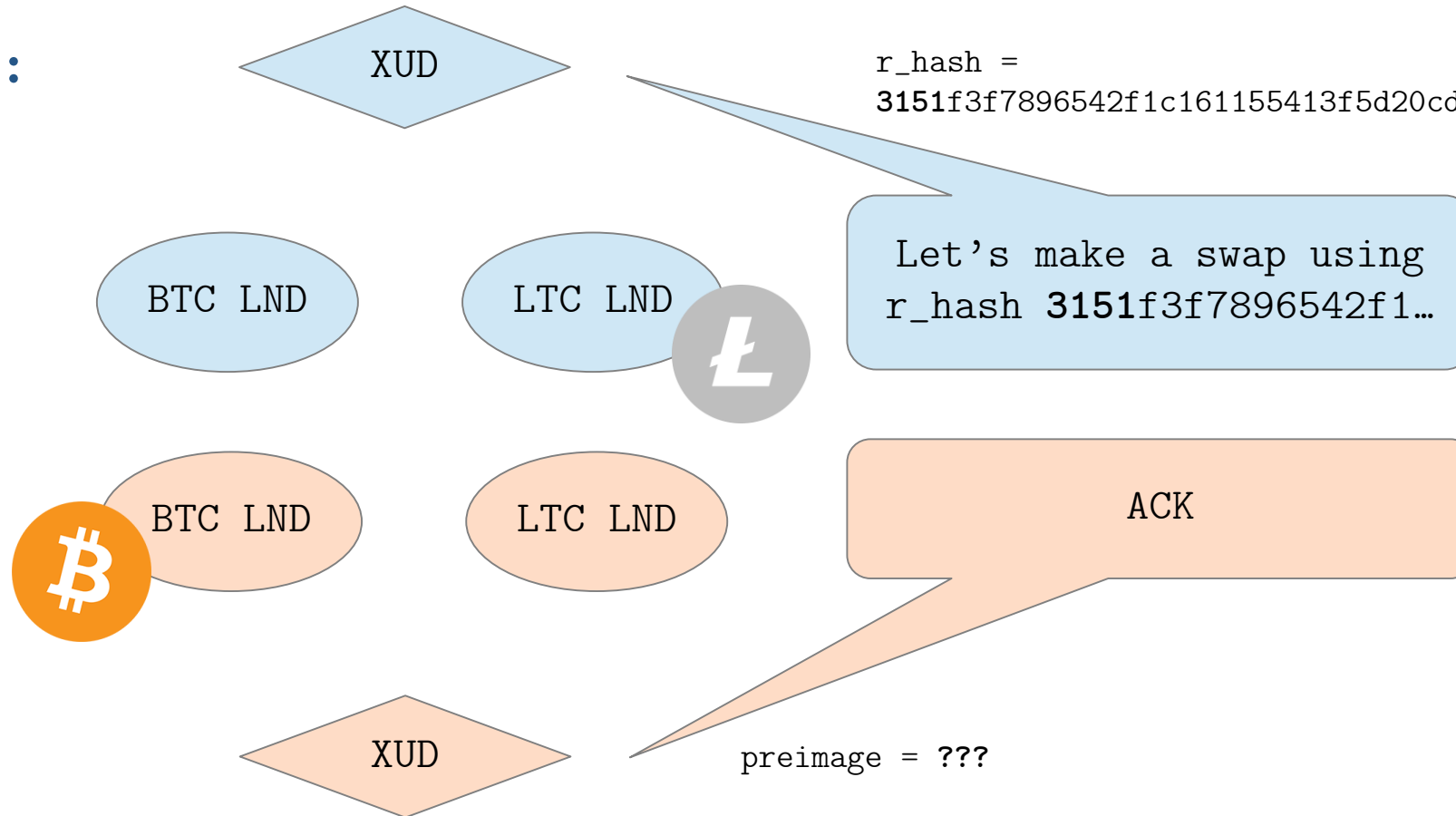
preimage = ???

r_hash =

3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

Atomic Swaps on ⚡

Step 2:



preimage = MVHz941lQvHBYRVUE/XSDNbzdrHSN14f3pANIhin6ZA=

r_hash =
3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

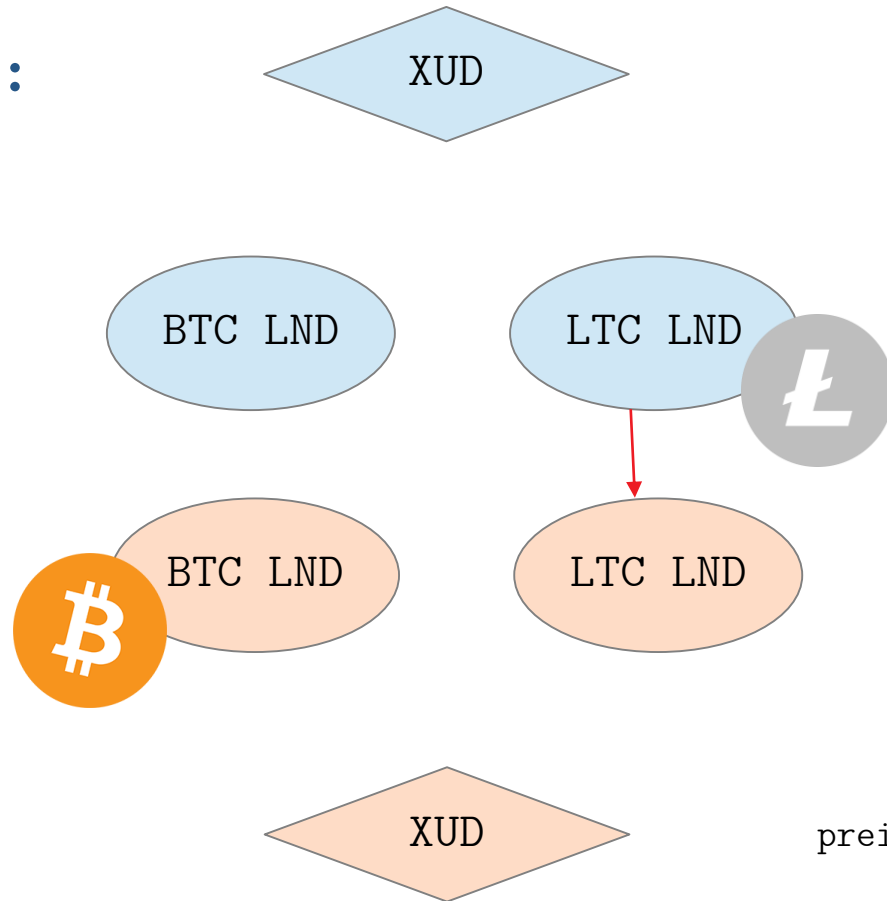
Let's make a swap using
r_hash 3151f3f7896542f1...

preimage = ???

r_hash =
3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

Atomic Swaps on ⚡

Step 2:



preimage = MVHz941lQvHBYRVUE/XSDNbzdrHSN14f3pANIhin6ZA=

r_hash =
3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

Pending HTLC

amount = 100 LTC
r_hash = 3151f3f7896542f1...

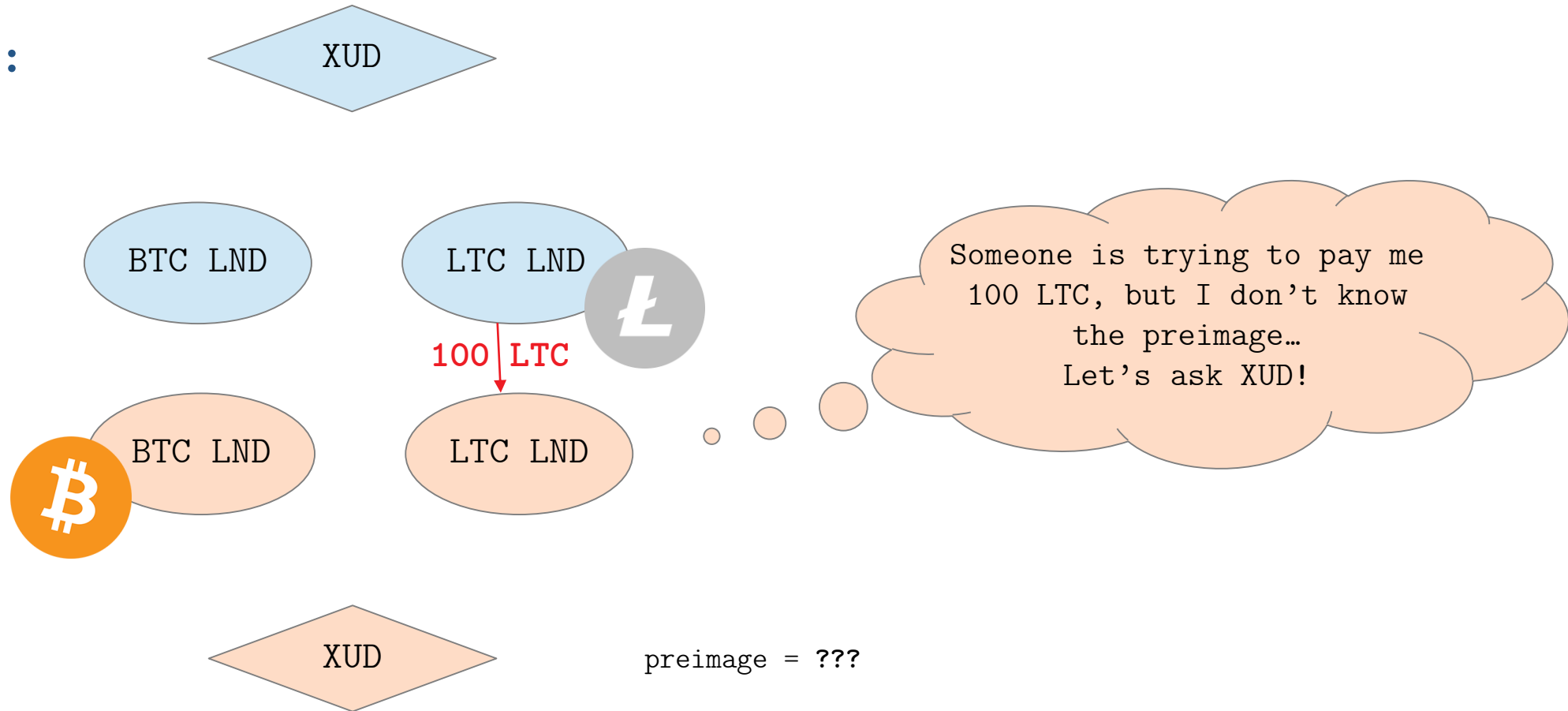
Note: This payment can route through
any number of intermediary nodes on
the LTC lightning network

preimage = ???

r_hash =
3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

Atomic Swaps on ⚡

Step 3:



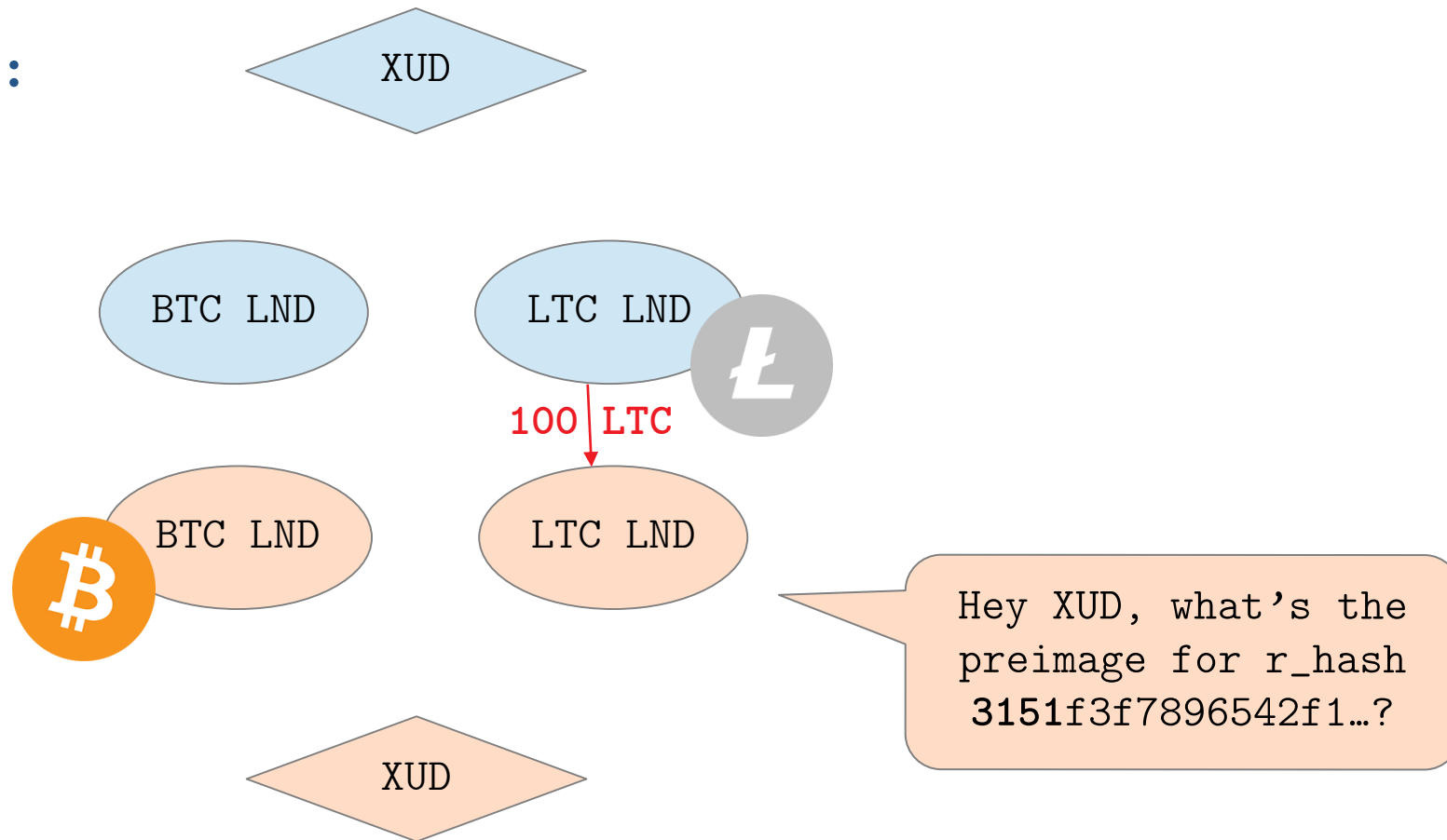
preimage = ???

r_hash =

3151f3f7896542f1c161155413f5d20cd6f376b1d2365e1fde900d2218a7e990

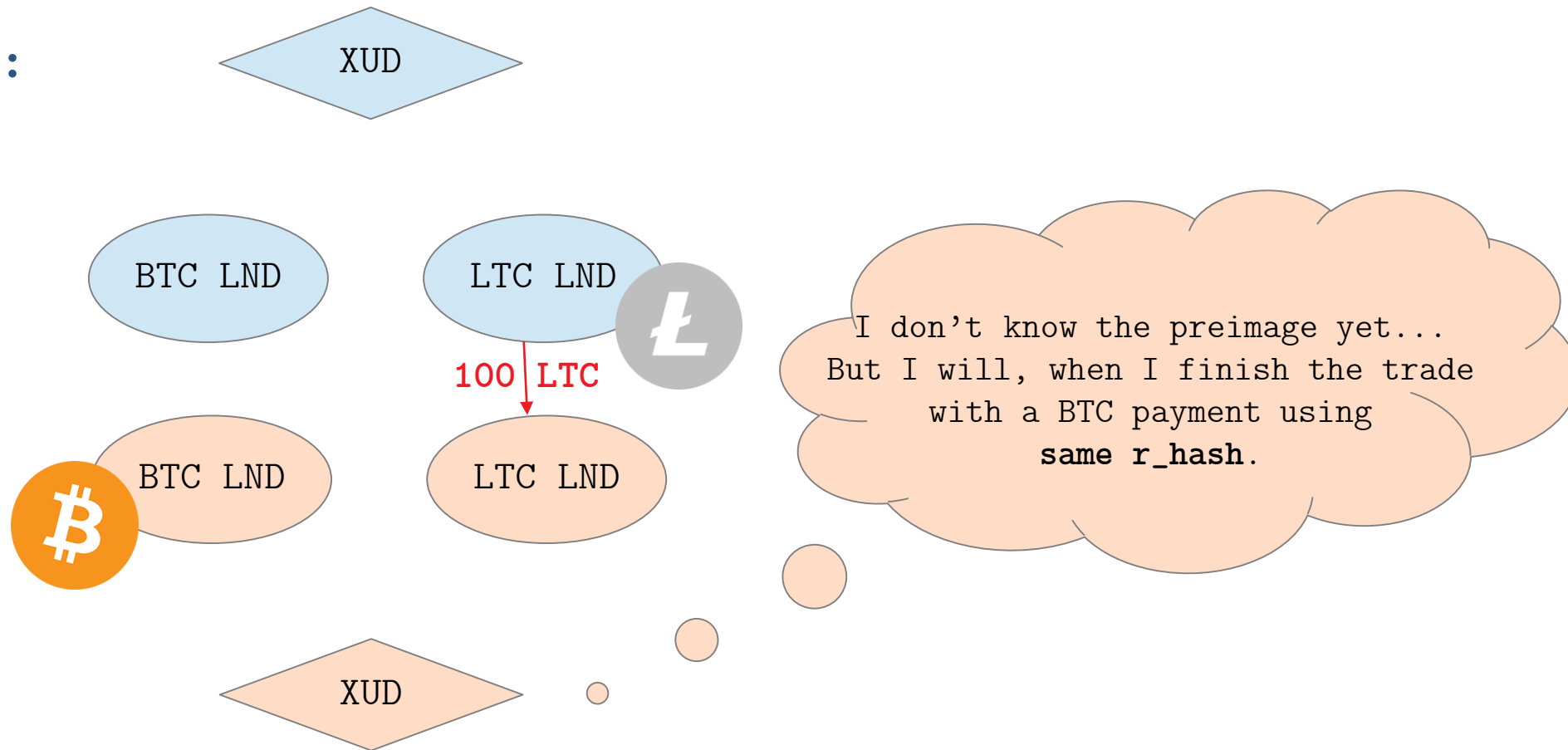
Atomic Swaps on ⚡

Step 3:



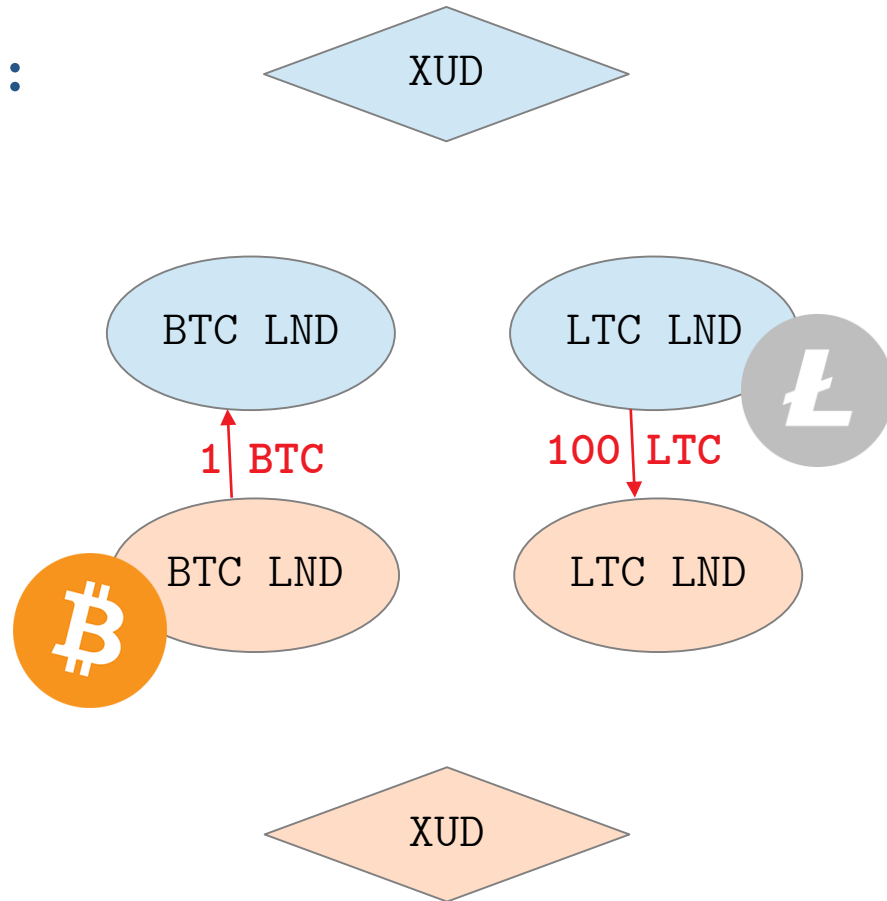
Atomic Swaps on ⚡

Step 4:



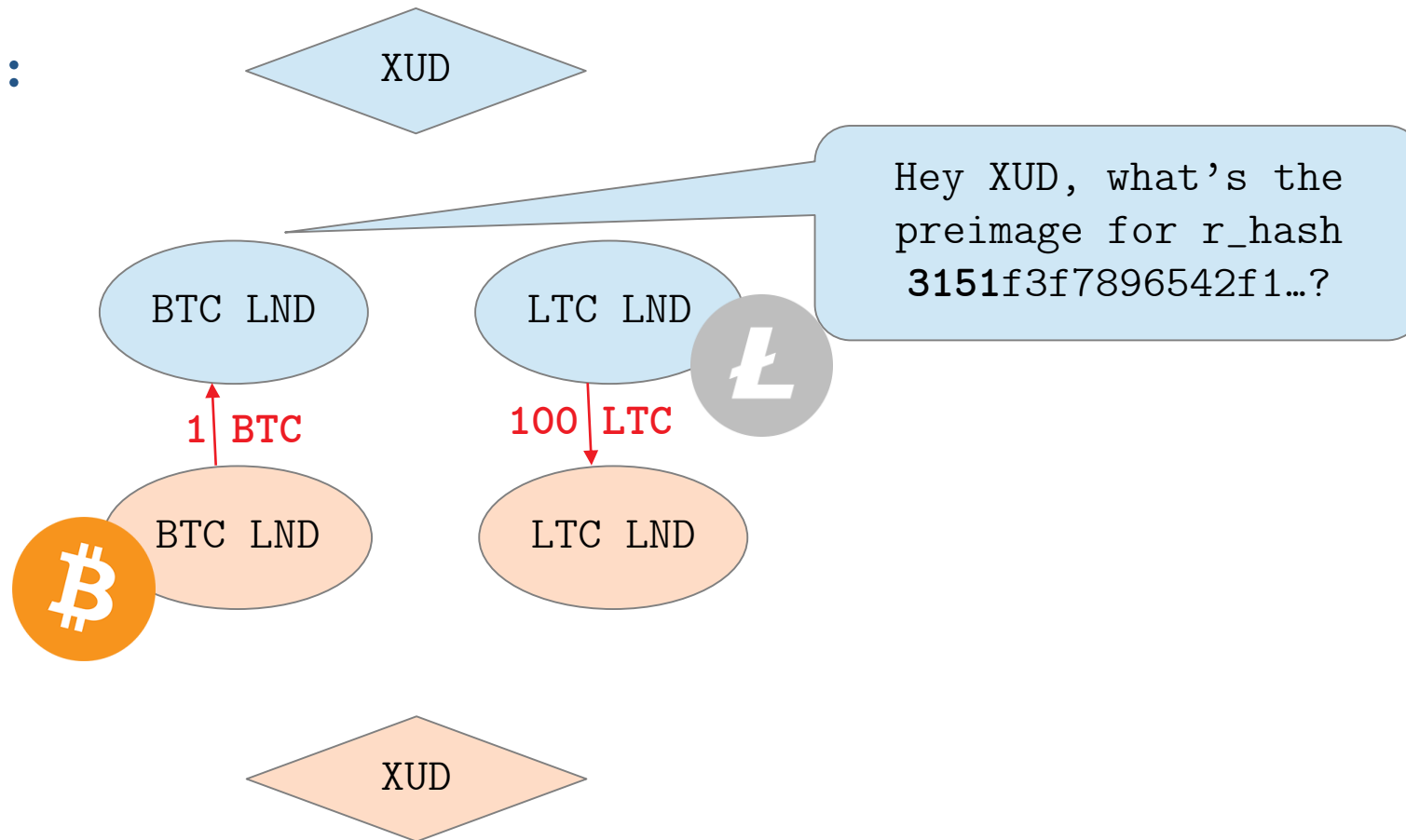
Atomic Swaps on ⚡

Step 4:



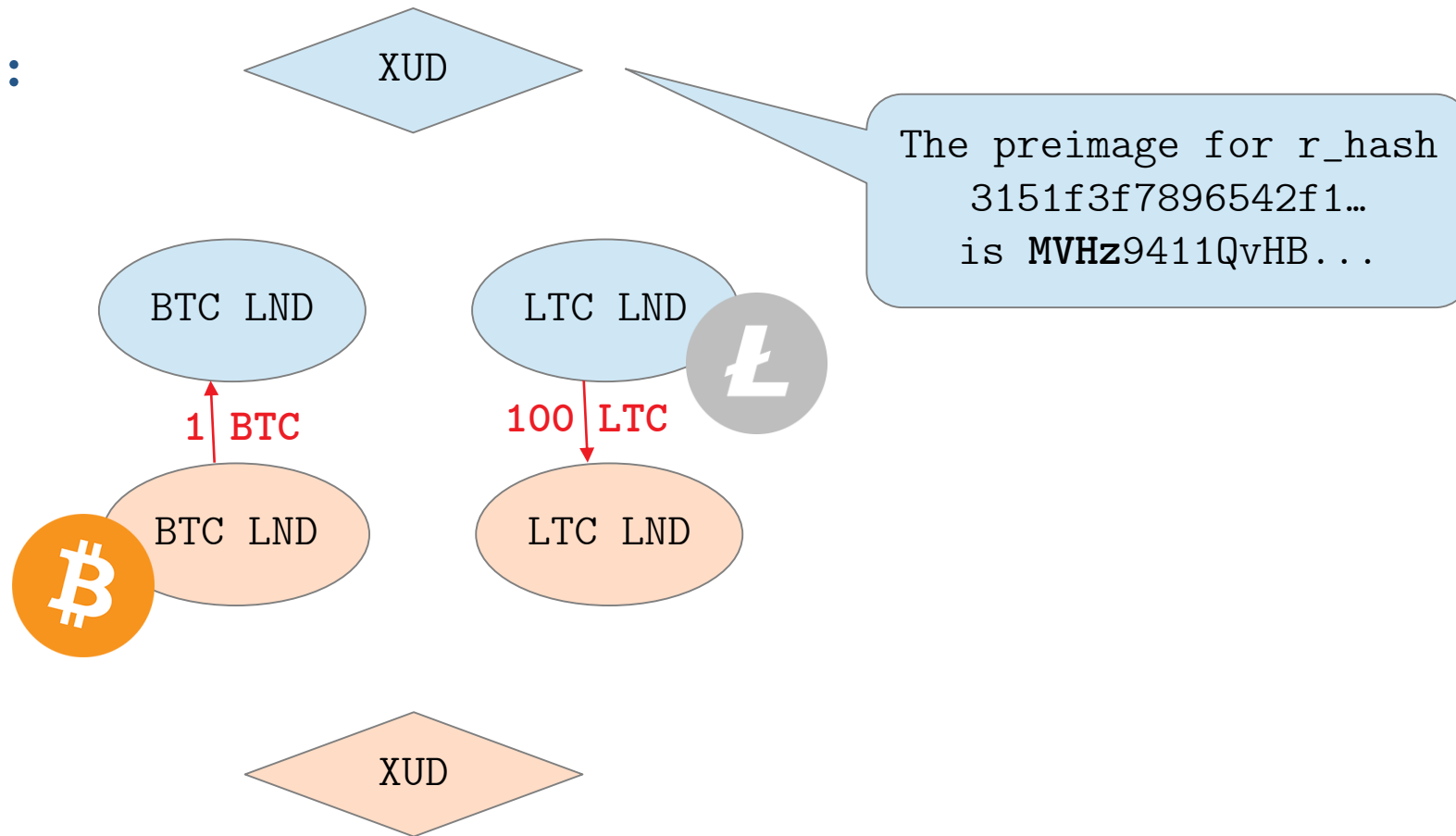
Atomic Swaps on ⚡

Step 5:



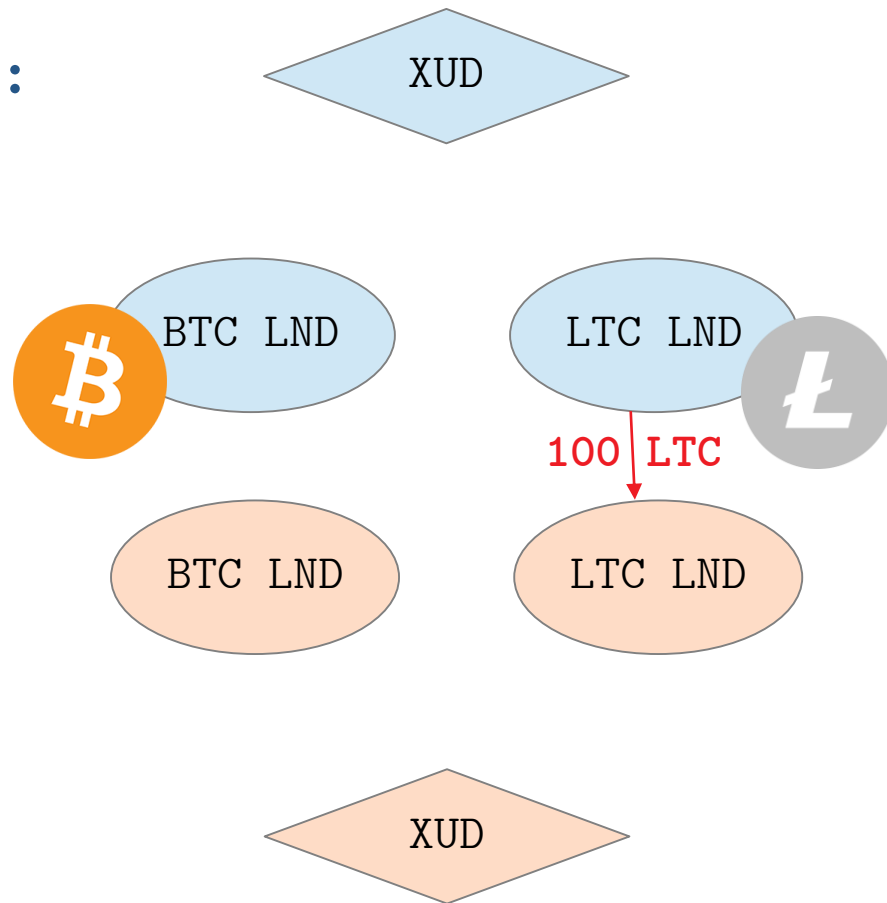
Atomic Swaps on ⚡

Step 6:



Atomic Swaps on ⚡

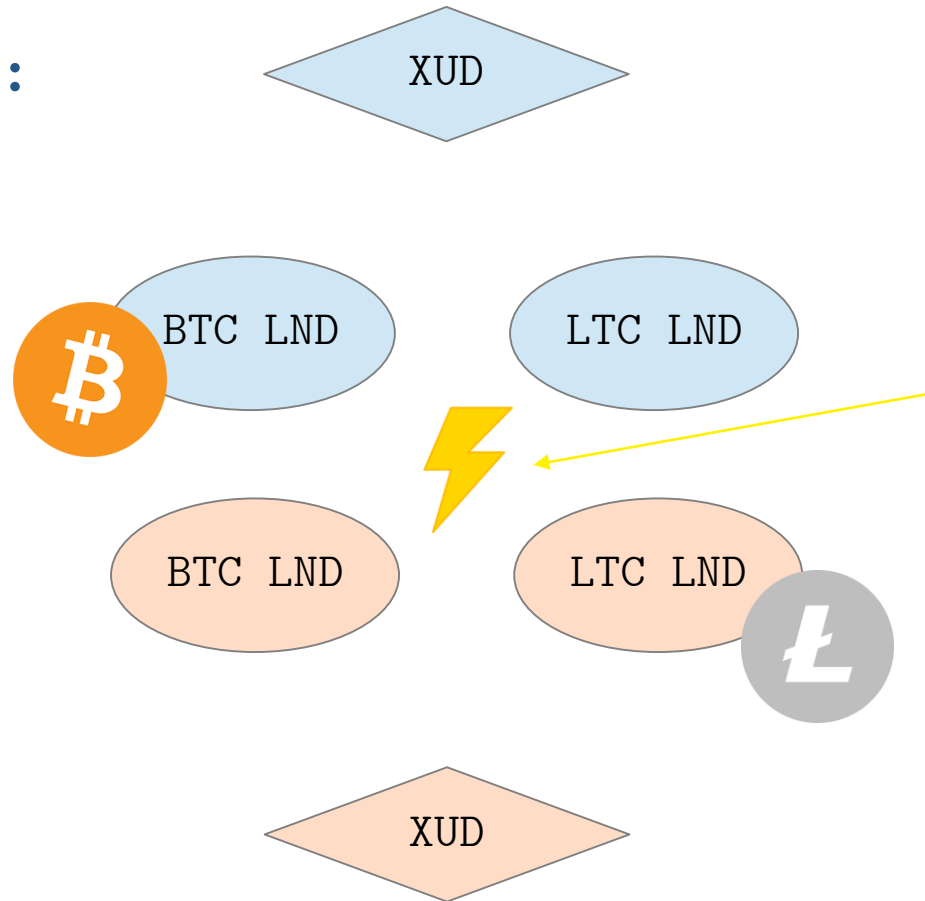
Step 6:



Preimage revealed

Atomic Swaps on ⚡

Step 6:



Preimage revealed

BTC & LTC Lightning payment settled!

Note: This payment can route through any number of intermediary nodes on the BTC lightning network

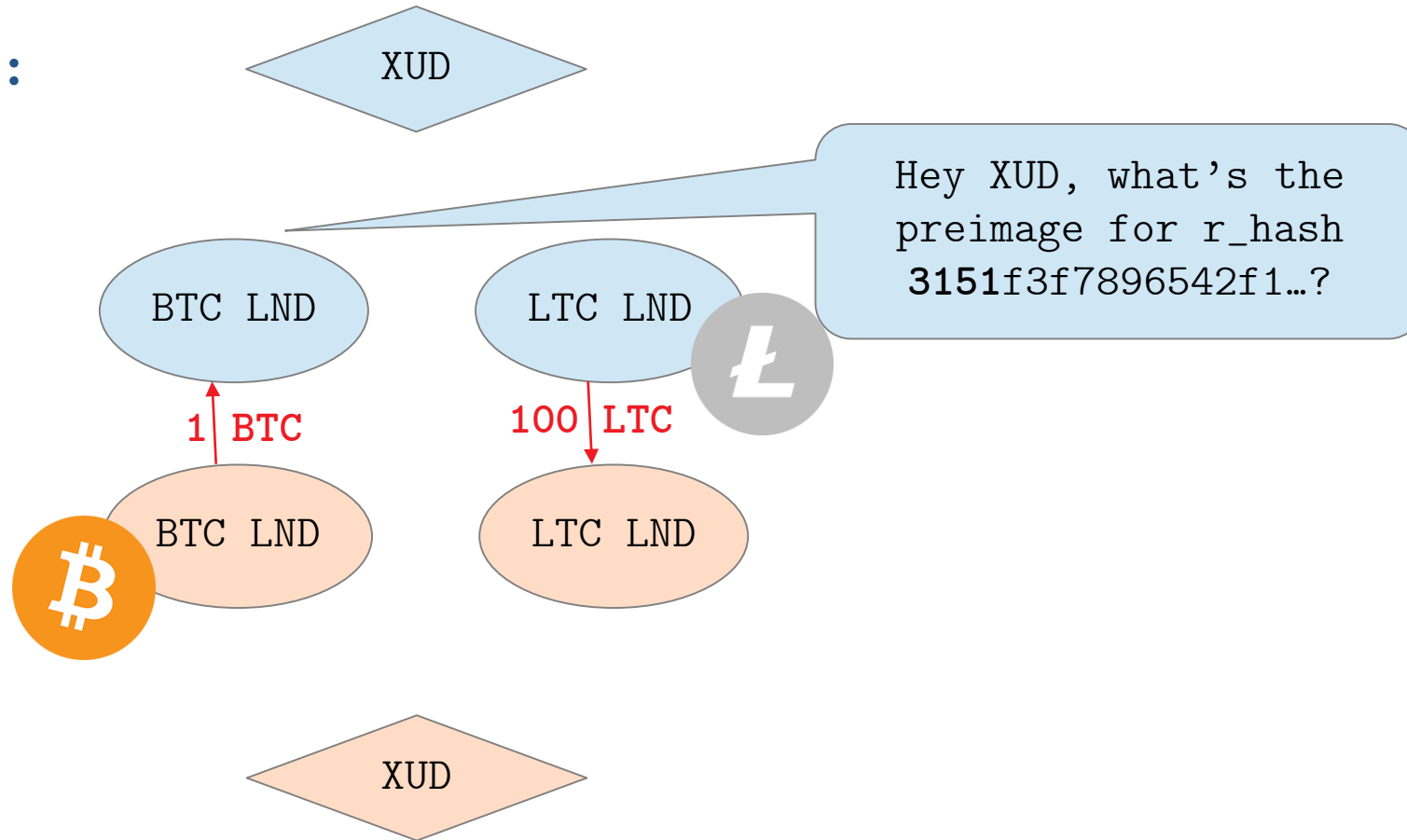
DIY: github.com/ExchangeUnion/xud/wiki/SimNet-Guide



But there is a BUT

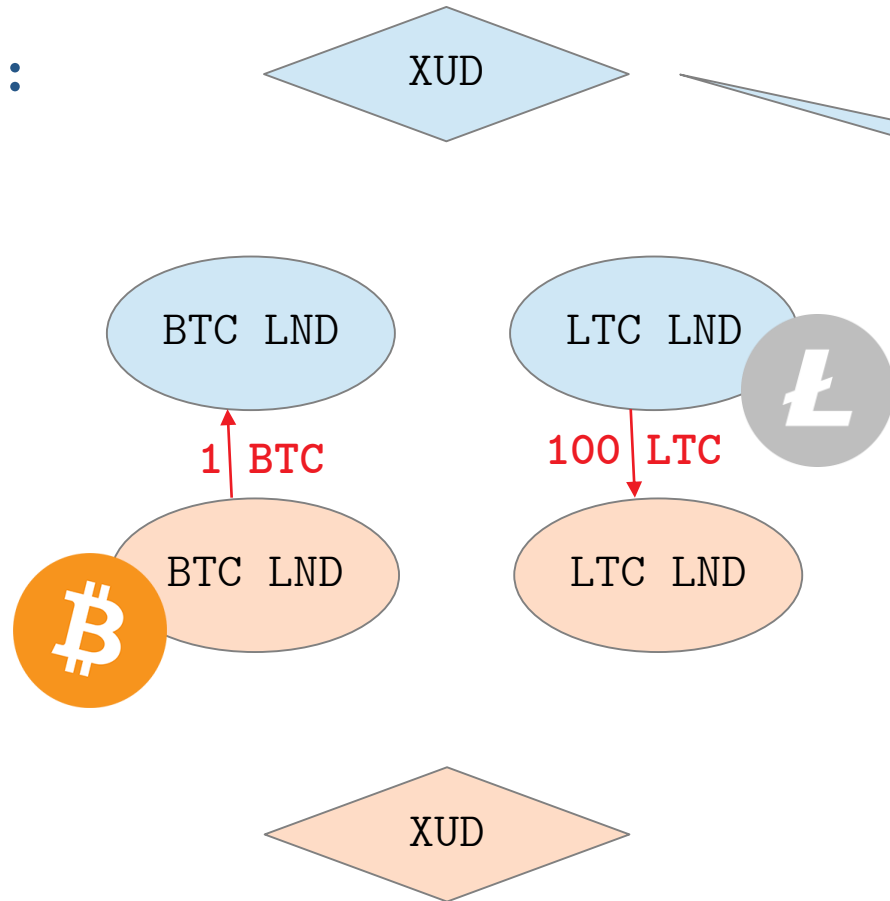
But there is a BUT

Step 5:



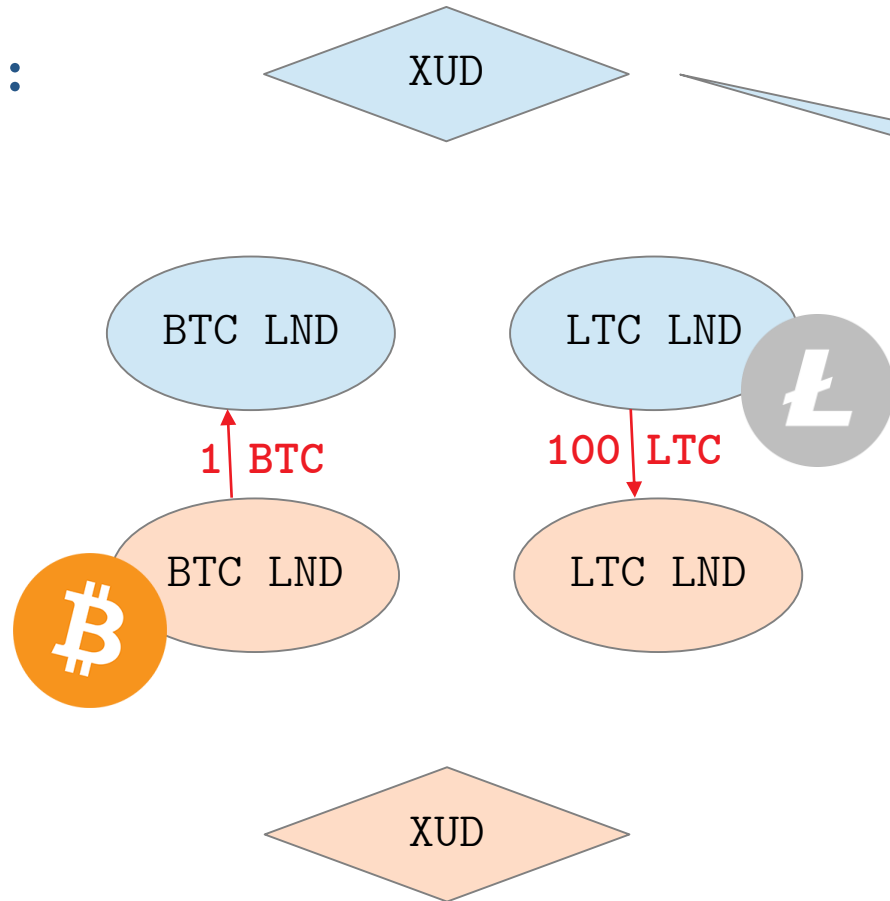
But there is a BUT

Step 5:



But there is a BUT

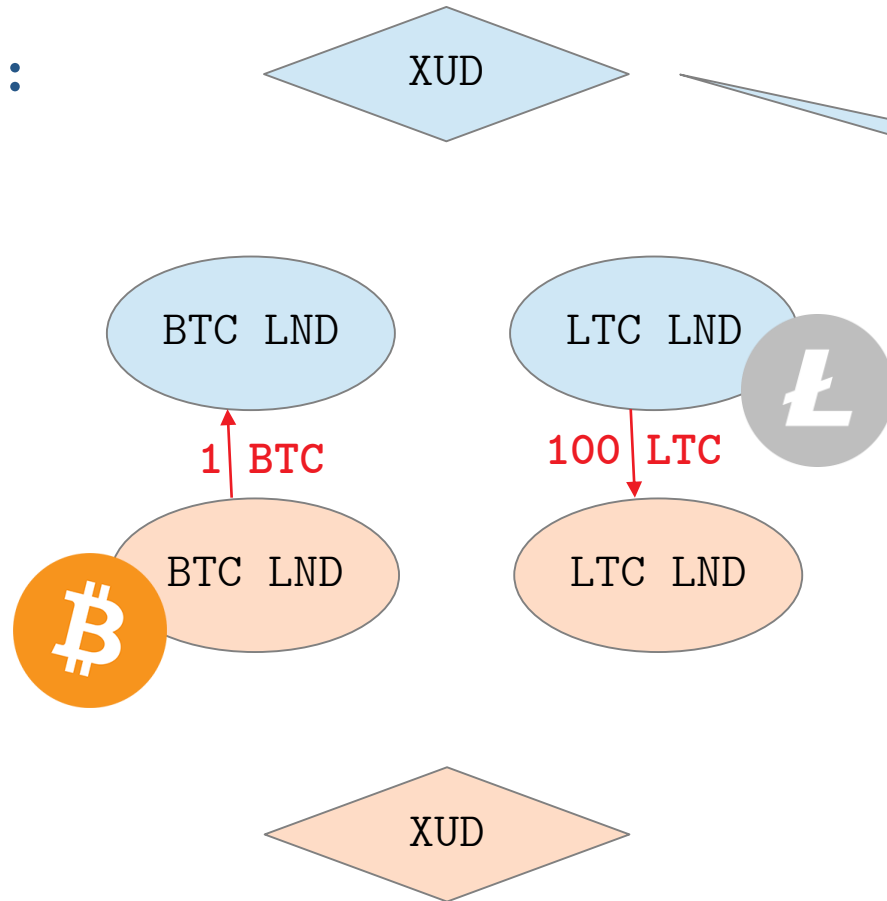
Step 5:



Up to timeout (min hours, up to one week)

But there is a BUT

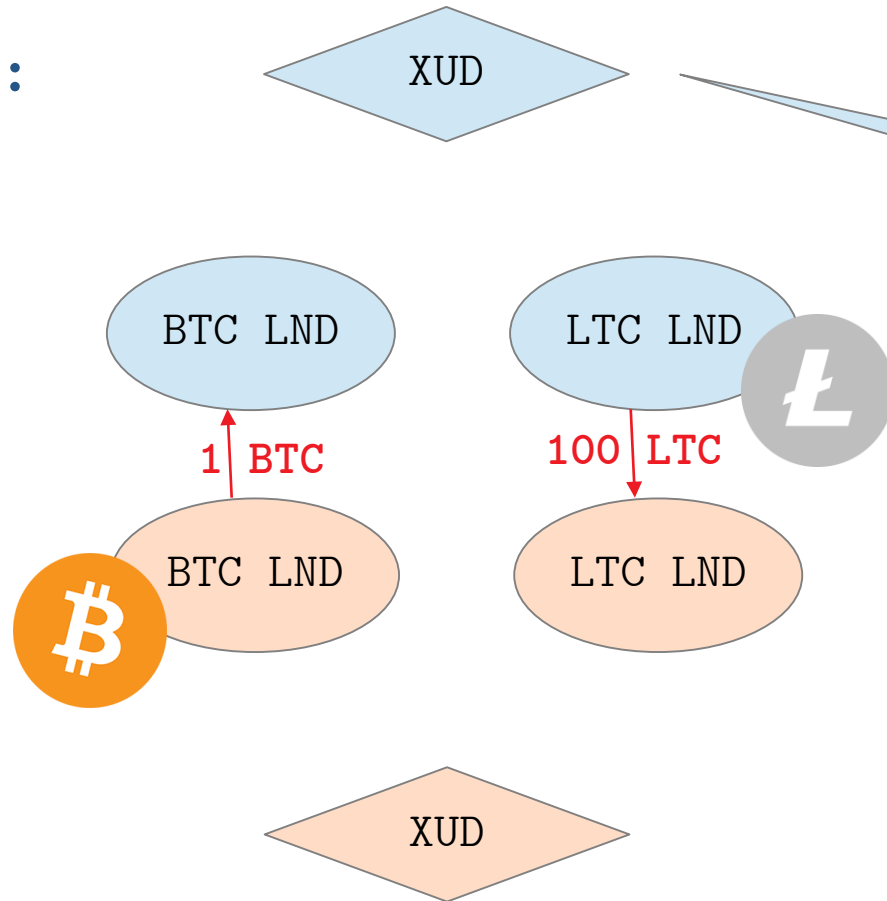
Step 5:



IF LTC vs. BTC went up
wait for timeout to get LTC back

But there is a BUT

Step 5:



IF LTC vs. BTC went up
wait for timeout to get LTC back
ELSE
release preimage

But there is a BUT

Option: right to execute a
trade or not in the future

But there is a BUT

Option: right to execute a
trade or not in the future

The problem: it's for free

But there is a BUT

Option: right to execute a trade or not in the future

The problem: it's for free

“Free Option Problem”



@kilrau

Free Option Problem

Good news: Conceptually solved
for on-chain atomic swaps

Free Option Problem

Good news: Conceptually solved
for on-chain atomic swaps

Bad News: Still unsolved for
lightning-based atomic swaps

Free Option Problem On-Chain

Conceptually solved. How?

Free Option Problem On-Chain

Punishment.

Free Option Problem On-Chain

Punishment.

Both parties A & B stake collateral in target asset in additional HTLCs which use the same preimage as in atomic swap.



@kilrau

Free Option Problem On-Chain

Punishment.

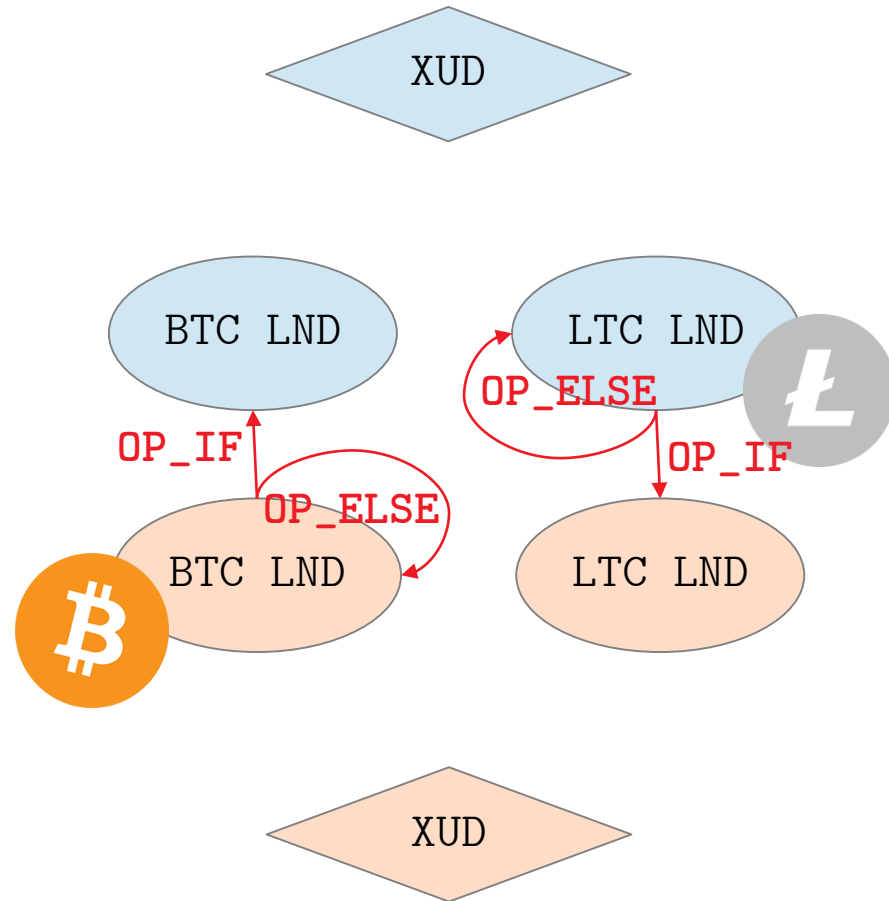
Both parties A & B stake collateral in target asset in additional HTLCs which use the same preimage as in atomic swap.

Both get collateral back, if preimage is released timely.

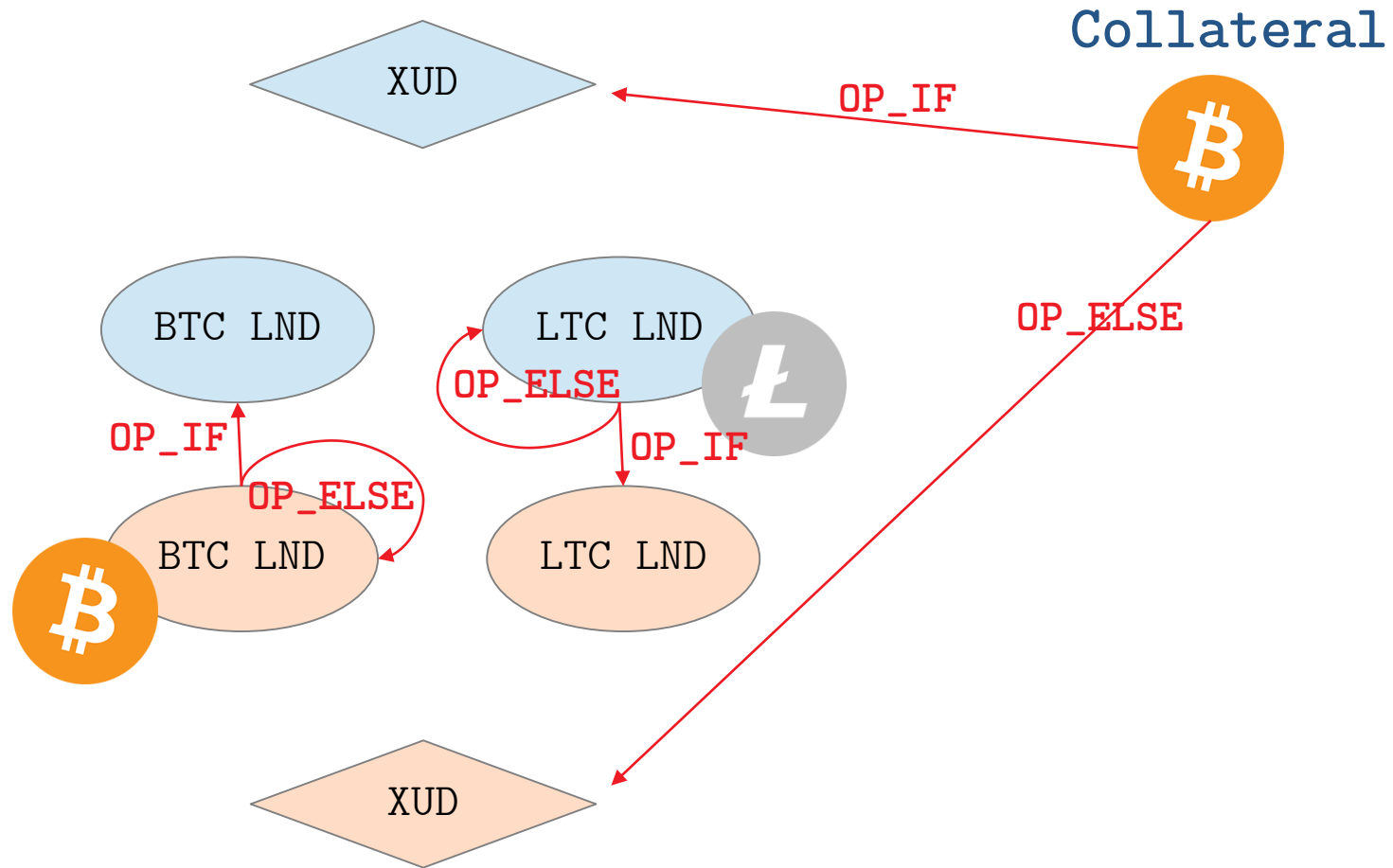


@kilrau

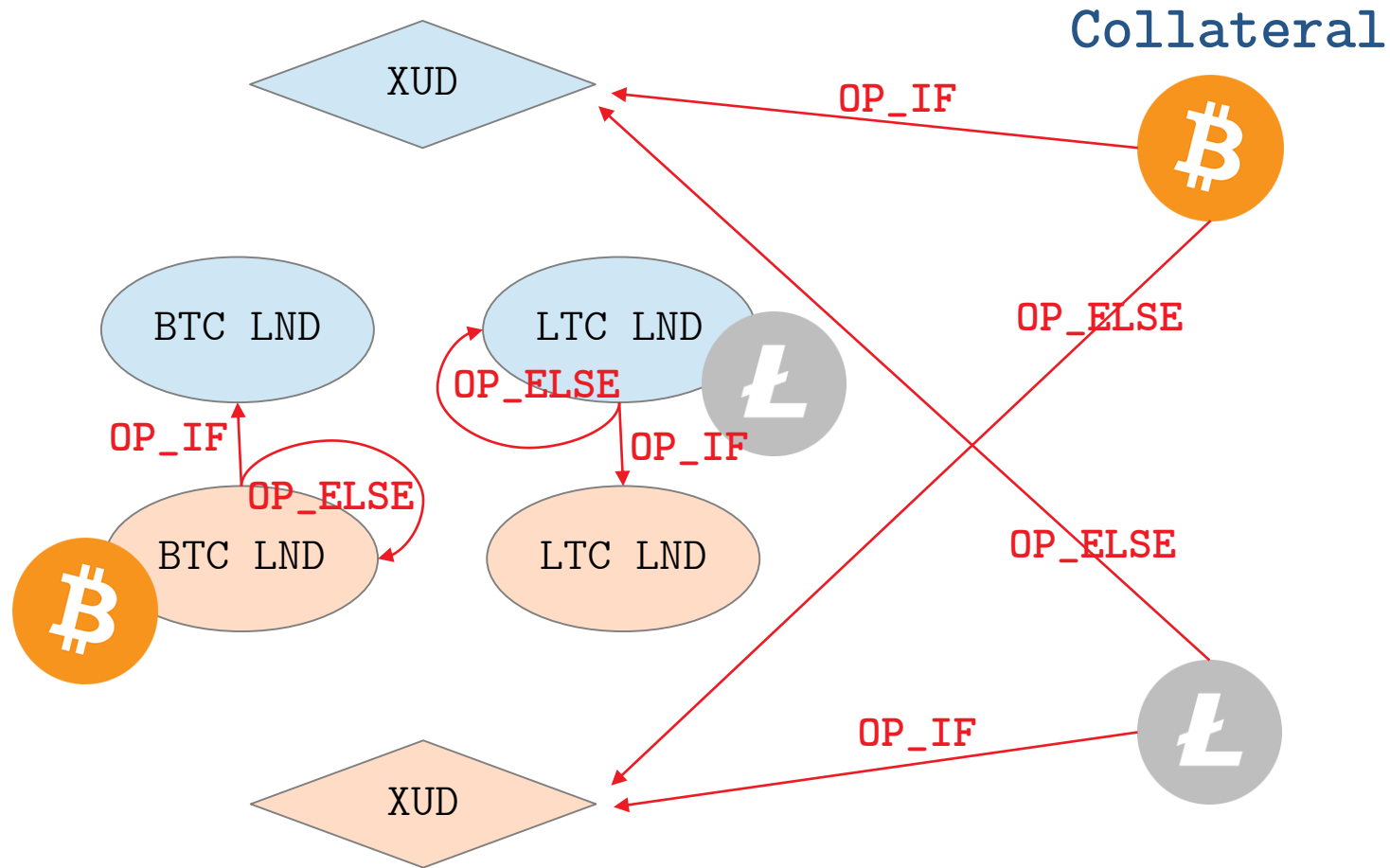
Free Option Problem On-Chain



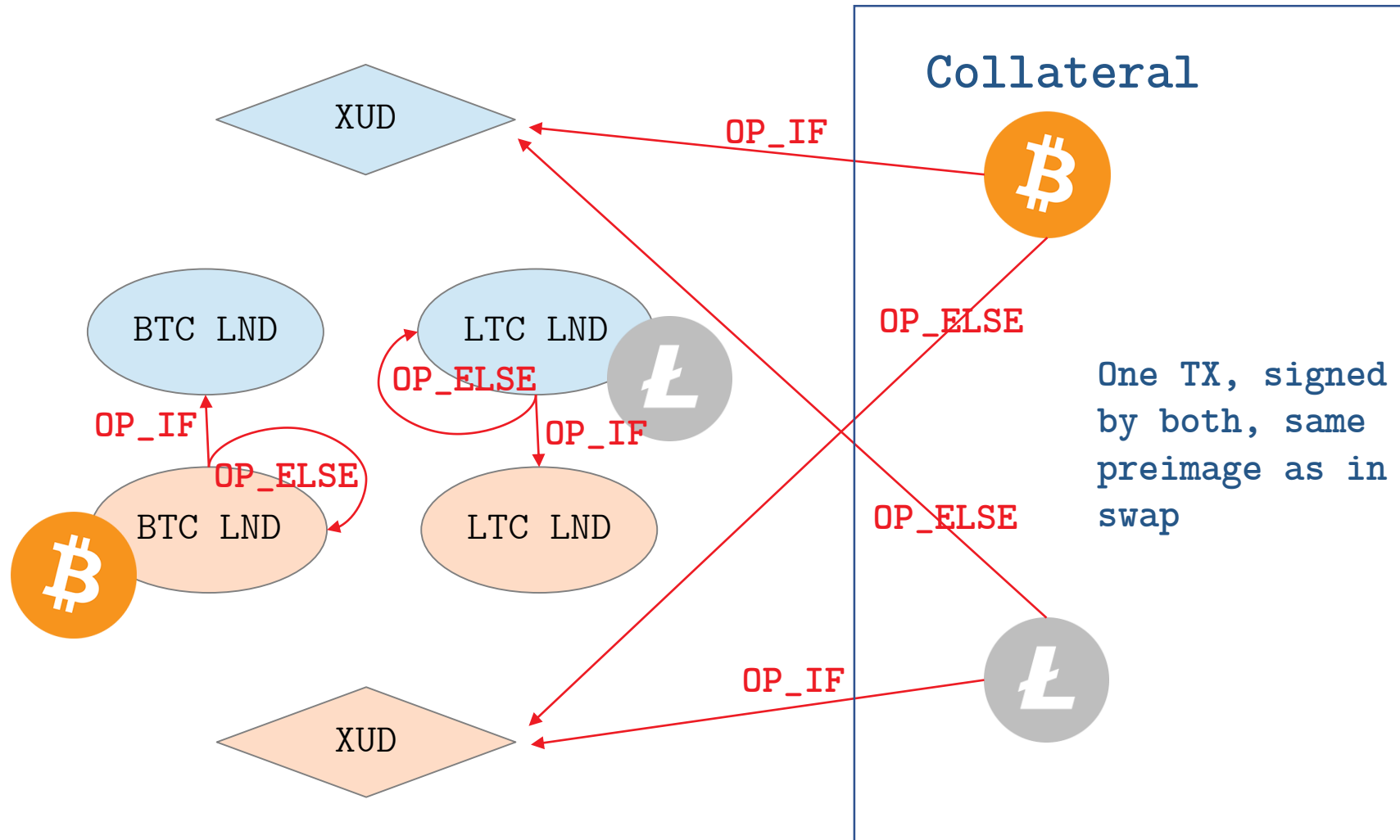
Free Option Problem On-Chain



Free Option Problem On-Chain



Free Option Problem On-Chain



Free Option Problem on LN

Why not do the same on LN?

Free Option Problem on LN

Problem:

- Uniquely attribute fault

Free Option Problem on LN

Problem:

- Uniquely attribute fault
- Needed: effective timeout of e.g. 10 seconds.
Not hours or days.

Free Option Problem on LN

Problem:

- Uniquely attribute fault
- Needed: effective timeout of e.g. 10 seconds.
Not hours or days.
- <https://lists.linuxfoundation.org/pipermail/lightning-dev/2018-December/001752.html>



@kilrau

Open Research Question

Join: gitter.im/exchangeunion/FOP

Open Research Question

Join: gitter.im/exchangeunion/FOP

Slides: github.com/exchangeunion/docs

Open Research Question

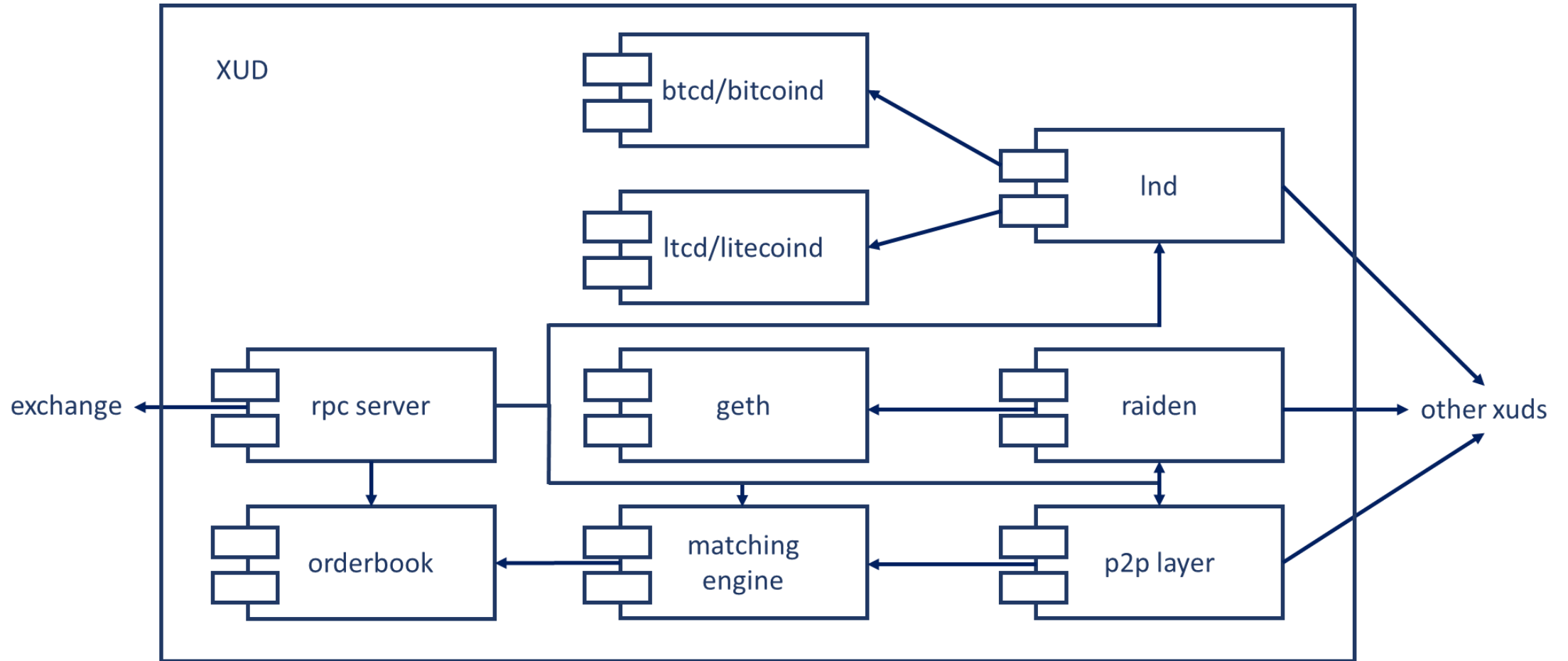
Join: gitter.im/exchangeunion/FOP

Slides: github.com/exchangeunion/docs

THX !

Backup

XUD, The Software



XUD, The Software

