

Trustless ONE<>BTC Bridge

[By [Ganesha Upadhyaya](#)]

This document describes the motivation, design, and components of the trustless ONE<>BTC bridge based on the interlay's XCLAIM approach.

Github: <https://github.com/harmony-one/onebtc> (does not have much yet, just a place-holder)

Motivation

Why onebtc?

- BTC volume and users still missing out on DeFi and other utilities
 - Centralized exchange backed wrapping, renBTC, tBTC together could only wrap 150K of 18M volume, which is 0.75% (<https://btconethereum.com>) . Even the wrapped 150K BTCs have low utility (in terms of daily transaction volume). On avg, 20K BTC daily transaction volume (<https://etherscan.io/token/0x2260fac5e5542a773aa44fbcfedf7c193bc2c599#tokenAnalytics>)
- BTC yet to see cheap, fast, and seamless DeFi
 - BTC hasn't found the utility (especially in DeFi) in spite of several attempts to make BTC available on Ethereum due to the *bridging cost and time* as well as the *cost and slowness of the utilities around wrapped BTC* due to Ethereum's inherent limitations.
- Industry trend
 - Many competitive chains are heading in this direction. E.g., PolkaDOT, Avalanche, nomic for cosmos, etc, in spite of having bridges to Ethereum. Thorchain enabled bitcoin liquidity before ethereum.
- Capturing the market
 - PolkaBTC is expected to be released sometime in Q1, 2021. Avalanche's ETH+BTC bridge is expected this quarter.
 - We want to have a strong show by the end of Q1/Q2, 2021 with trustless, cheap, fast, and seamless bridges to both Ethereum and BTC to give us a competitive advantage for progressing our DeFi utilities throughout 2021.
- Possible partnerships with BTC altcoins

- [See [Stephen's blog](#) on our broad vision of cross-chain pools.]

OneBTC vs Interlay

- PolkaBTC parachain won't come to life until mid this year
- Interlay wants to target EVM chains.
- They approached us with a co-fund project opportunity. I am encouraging them with "build on harmony, and reuse the bridge modules for other EVM chains".

Why interlay xclaim approach for onebtc, why not tBTC or renBTC?

- Timeline
 - We have explored all the three options and the most viable option in terms of the timeline is onebtc (using xclaim approach), i.e., partnering with interlay
- Partnership priorities
 - tBTC is prioritizing value chains first (where users, devs, and utilities exists already), renBTC's multichain approach is most suitable for Ethereum-like chains (e.g., Ethereum, Binance Smart Chain) due to tooling limitations
 - We may still have to integrate Harmony with tBTC infrastructure to attract more liquidity (as tBTC is hoping to add multiple chains to their bridge infrastructures to maximize the benefits for bridge validators as well as keep the bridge cost for users reasonable). Also, work with renBTC to abstract their tooling layer to accommodate non-Ethereum like chains from the tooling perspective.
- Feasibility
 - Xclaim approach has least dependencies. Further, most components can be built out independently and tested to make gradual progress. All components of the system are going to be trustless with minimal incentivization needs, when compared to fully validator network based approaches like tBTC and renBTC

Utilities for onebtc? Similar to PolkaBTC's shown below:

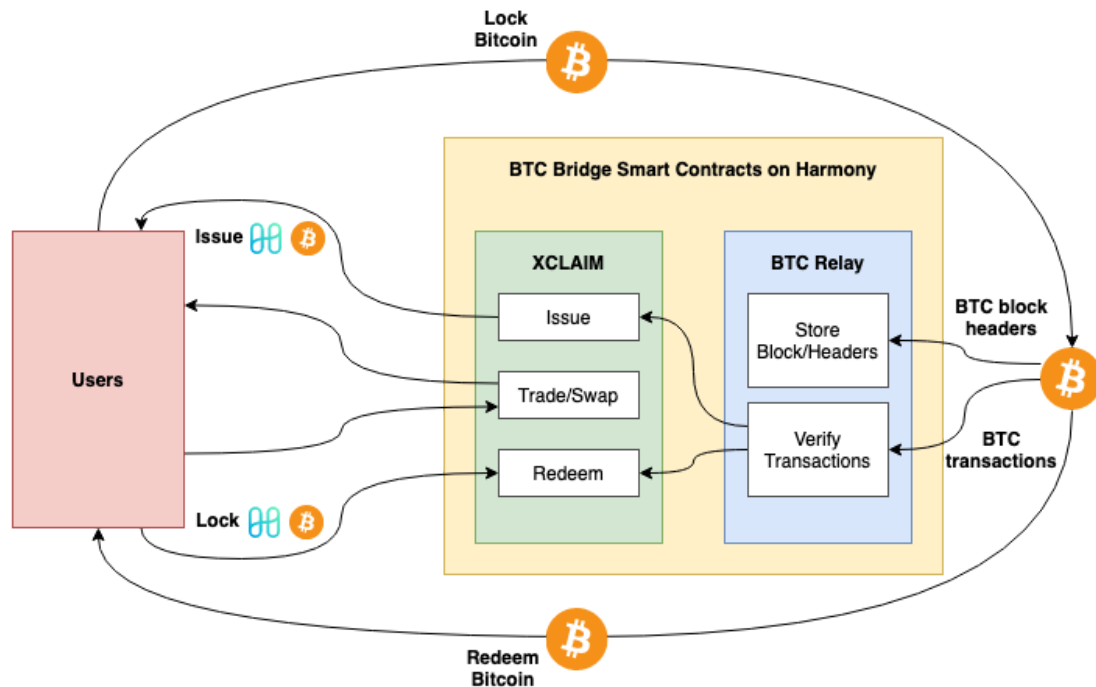
Planned Integrations

The Interlay team has started working with teams building on Polkadot to enable early access to PolkaBTC:

- **Acala** and Interlay are exploring different collateral use cases between Acala's aUSD stablecoin and PolkaBTC,
- **Laminar** Laminar is exploring collaboration with Interlay to provide cross-chain liquidity to Polkadot ecosystem,
- **Plasm Network** is looking to offer PolkaBTC to its highly scalable DApps,
- **Equilibrium** is planning to offer PolkaBTC on cross-chain DeFi money markets,
- **Edgeware** is looking to offer support for PolkaBTC in its WASM smart contracts,
- **Moonbeam** (developed by PureStake) is planning to use PolkaBTC with its Ethereum compatible smart contract platform,
- **Polkaswap** is to support trading PolkaBTC on an AMM-powered decentralized exchange,
- **Chorus One** will explore how to make PolkaBTC available to the **Cosmos SDK** ecosystem via a Cosmos-Substrate IBC bridge.

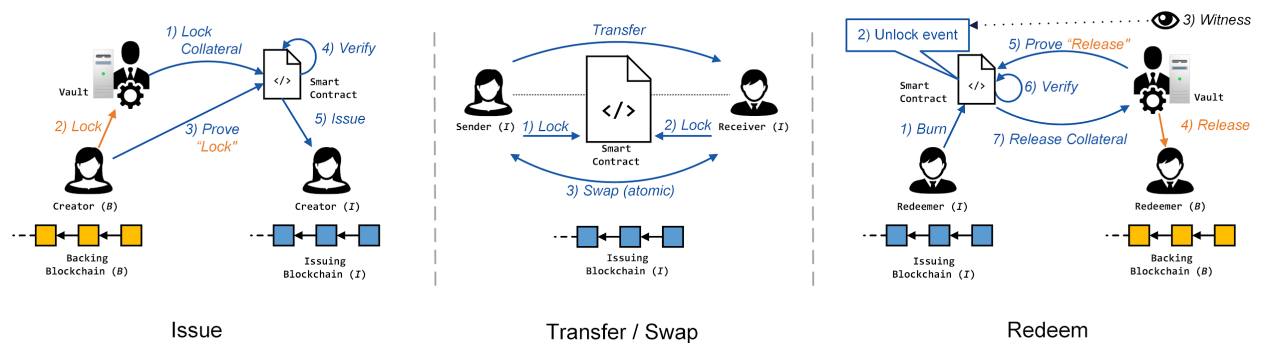
If you're building a project on top of Polkadot and want to integrate with PolkaBTC, visit polkabtc.io or reach out via polkabtc@interlay.io.

Overview



Ref: <https://onebtc-dev.web.app/intro/at-a-glance.html>

XCLAIM protocol



Ref: <https://interlay.gitlab.io/polkabtc-spec/intro/CbA.html>

- **Issue:** process allows a user to lock Bitcoin on the Bitcoin chain and, in return, issue 1BTC on Harmony
- **Transfer/swap:** process allows a user to transfer 1BTC to others within Harmony

- **Redeem:** process allows a user to burn 1BTC on Harmony and redeem previously locked Bitcoins on Bitcoin

Actors

1. Vaults

- Collateralized intermediaries that are active on both the backing blockchain (Bitcoin) and the issuing blockchain to provide collateral in ONE.
- They receive and hold BTC from users who wish to create 1BTC tokens.
- When a user destroys 1BTC tokens, a Vault releases the corresponding amount of BTC to the user's BTC address.
- Vaults interact with the following modules directly: Vault Registry, Redeem, and Replace.

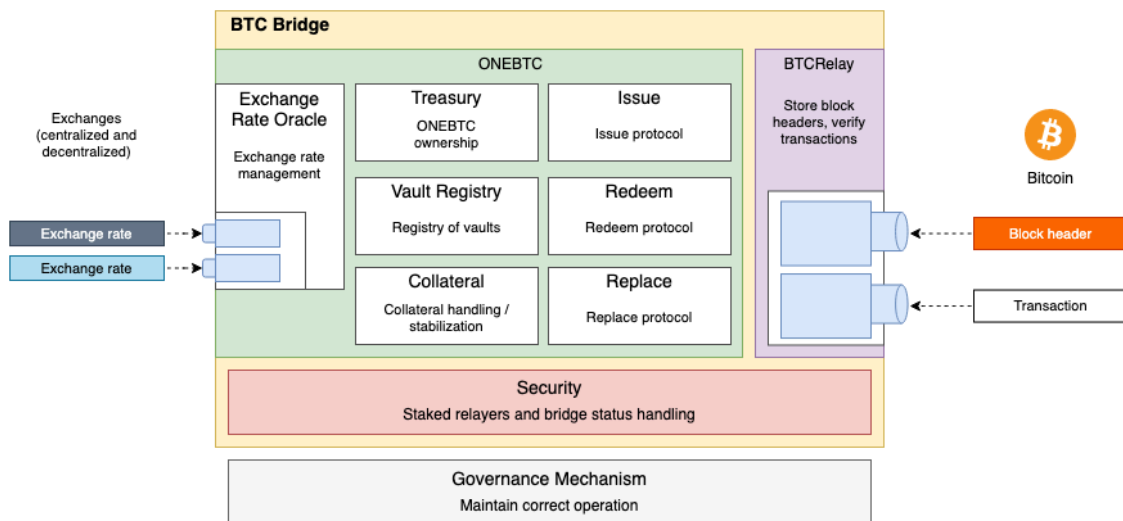
2. Users

- Interact with the BTC Bridge smart contracts to create, use (trade/transfer/...), and redeem Bitcoin-backed 1BTC tokens
- Sub-roles
 - Requester: A user that locks BTC with a Vault on Bitcoin and issues 1BTC on Harmony. Interacts with the Issue module.
 - Sender and Receiver: A user (Sender) that sends 1BTC to another user (Receiver) on Harmony. Interacts with the Treasury module.
 - Redeemer: A user that destroys 1BTC on Harmony to receive the corresponding amount of BTC on the Bitcoin blockchain from a Vault. Interacts with the Redeem module.

3. Staked Relayers

- Collateralized intermediaries which run Bitcoin full nodes
 - Monitor validity and availability of transactional data for Bitcoin blocks submitted to BTC-Relay
 - Monitor that Vaults do not move locked BTC on Bitcoin without prior authorization by the BTC Bridge smart contract (i.e., through one of the Issue, Redeem or Replace protocols).
 - In case either of the above errors was detected, Staked Relayers report this to BTC Bridge smart contract.
- Interact with the BTC-Relay, Security, and Vault Registry modules.

Architecture



Ref: <https://onebtc-dev.web.app/intro/architecture.html>

Components or Modules

Btc-relay

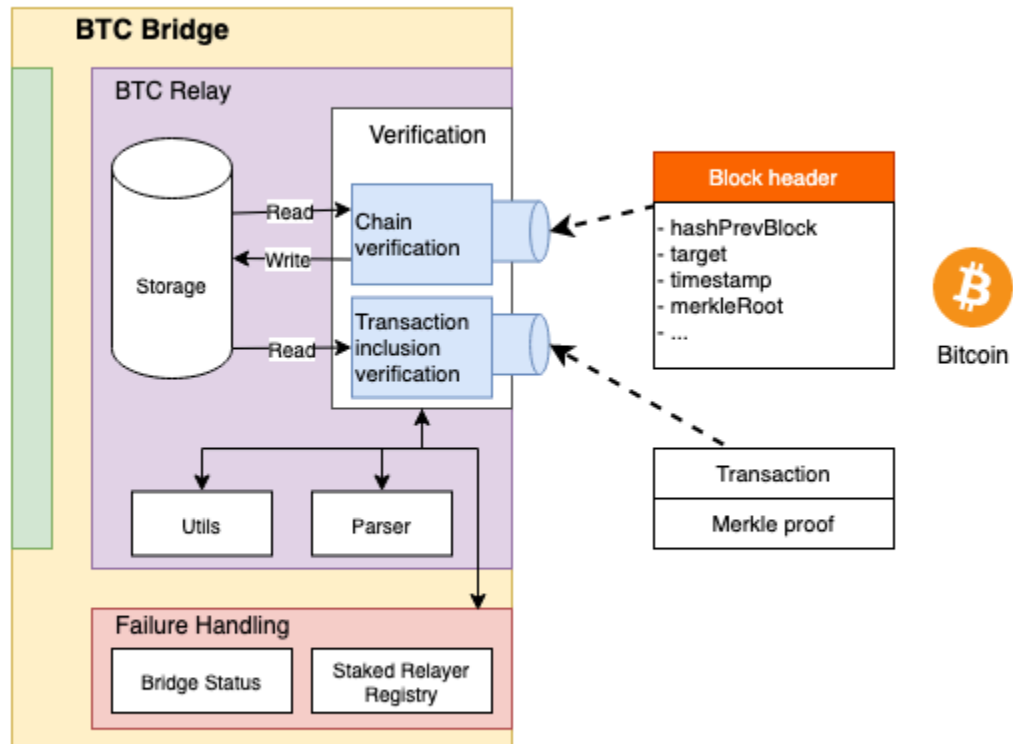
Functionality: verify the state of Bitcoin and react to transactions and events. Acts as Bitcoin SPV light client on Harmony, storing Bitcoin block headers and allowing users to verify transaction inclusion proofs. Further, it is able to handle forks and follows the chain with the most accumulated Proof-of-Work.

Ref: <https://onebtc-relay.web.app/intro/architecture.html>

Spec: <https://onebtc-relay.web.app/intro/architecture.html>

Impl:

- Rust code (recent):
<https://github.com/interlay/BTC-Parachain/tree/master/crates/btc-relay/src>
<https://github.com/interlay/BTC-Parachain/tree/master/crates/bitcoin/src>
- Old solidity: <https://github.com/crossclaim/btcrelay-sol/tree/master/contracts>



- We need to implement as solidity contract

Exchange Rate Oracle

- The Oracle module maintains the exchange rate value between the asset that is used to collateralize Vaults (ONE) and the to-be-issued asset (BTC).

Treasury

- The Treasury module maintains the ownership and balance of 1BTC token holders. It allows respective owners of 1BTC to send their tokens to other entities and to query their balance. Further, it tracks the total supply of tokens.

- *We may not need this module.*

Vault Registry

- The central place to manage vaults. Vaults can register themselves here, update their collateral, or can be liquidated.
- Similarly, the issue, redeem, and replace protocols call this module to assign vaults during issue, redeem, and replace procedures.

Collateral

- The central storage for collateral provided by users and vaults of the system.
- It allows to (i) lock, (ii) release, and (iii) slash collateral of either users or vaults.
- It can only be accessed by other modules and not directly through external transactions.

Issue

- The issue module allows a user to create (or mint) new 1BTC tokens.
- Steps: <https://interlay.gitlab.io/polkabtc-spec/spec/issue.html>

Redeem

- The redeem module allows a user to receive BTC on the Bitcoin chain in return for destroying an equivalent amount of 1BTC on the Harmony chain.
- Steps: <https://interlay.gitlab.io/polkabtc-spec/spec/redeem.html>

Replace

- The Replace module allows a Vault (OldVault) to be replaced by transferring the BTC it is holding locked to another Vault (NewVault), which provides the necessary ONE collateral.
- Steps: <https://interlay.gitlab.io/polkabtc-spec/spec/replace.html>

Security

- The Security module is responsible for tracking the status of the BTC Bridge on Harmony, flagging failures such as liveness and safety failures of BTC-Relay or crashes of the exchange-rate-oracle.
- Details: <https://interlay.gitlab.io/polkabtc-spec/spec/security.html>

PolkaBTC repos:

- Polkabtc frontend: <https://beta.polkabtc.io/>
 - Polkabtc app: <https://beta.polkabtc.io/app>
 - Polkabtc dashboard: <https://beta.polkabtc.io/dashboard> , repo (<https://github.com/interlay/polkabtc-stats>)
 - Frontend repo: <https://github.com/interlay/polkabtc-ui>
 - Polkabtc clients (vaults, relayers): <https://github.com/interlay/polkabtc-clients>
 - JS SDK: <https://github.com/interlay/polkabtc-js>

 - Polkabtc public docs: <https://docs.polkabtc.io/#/>
 - Polkabtc spec (latest): <https://gitlab.com/interlay/polkabtc-spec>
 - Champaigns to bootstrap bridge: <https://beta.polkabtc.io/challenges>
-


Ownership (❤️ Harmony, 🏠 Community)

- 🏠 0. Architecture: specification, requirements, documentation
- ❤️ 1.1 BTC-Relay (50% done, ETA 2/28)
- ❤️ 1.2 Bitcoin Library (20% done, ETA 2/20)
- ❤️ 1.3 Collateral (50% done, see BUSD and wrapped stablecoins)
- ❤️ 1.4 Exchange Rate Oracle (100% done; see our apps using Chainlink and Band)
- 🏠❤️ 1.5 Fee Model
- 🏠❤️ 1.6 Issue
- 🏠 1.7 Redeem
- 🏠 1.8 Replace
- 🏠 1.9 Security
- 🏠 1.10 Relayer Registry
- 🏠❤️ 1.11 Treasury
- 🏠 1.12 Vault Registry
- ❤️ 2. Client (see our [Coinbase's Rosetta API](#))
- ❤️ 3. Web UI (see our [Uniswap clone](#) and [ETH Bridge](#) UI)
- ❤️ 4. Testnets & Mainnets (Harmony mainnet live since 2019Q2, open staking since 2020Q2)
- ❤️ 5. Audit (with Peckshied)

Harmony's features

- Full compatibility of Solidity and EVM bytecode (see our [showcases](#) and [6 hackathons](#) with Gitcoin)
- Full compatibility of Metamask and web3.js (or any Ethereum toolings, wallets and portals)
- Ethereum bridge (based on Flyclient; see [research whitepaper](#) with Mahdi Zamani)
- Cross-chain Uniswap-fork to swap assets from Harmony (done in 2020Q3), Ethereum (done in 2020Q4), Binance Smart Chains (ETA 2021/Q1), Bitcoin (ETA 2021/Q2)
- Cross-chain finance products such as lending and pools, per [Harmony 2021 strategy](#)

See <https://harmony.one/bridge> and <https://harmony.one/layer2>.

**stephen tse | s.one**
@stse

our research paper on trustless bridges with [@mahdi_zamani](#) is accepted!

join our talk on 3/9 tuesday 1pm pst


preprint at harmony.one/bridge-research

downloads and verifies every block header. A light client of a Proof-of-Stake blockchain only needs to verify the signatures of the validators who signed the blocks. Since the validator set does not change within an epoch (which is nearly 1 day), the light client only needs to download 1 block header that contains the validator set per epoch. This in itself reduces the light client complexity substantially (1/32768 blocks).

Next, the BLS signature aggregation. Harmony's consensus mechanism requires a quorum of 2/3 or more validators to sign each block to commit it to the blockchain. Harmony validators use BLS signatures to sign the blocks. Instead of having each of these signatures be separately stored on-chain, one aggregated signature is stored on-chain—a nice property of BLS, which allows such aggregation. This further reduces the work that the light client has to perform for verifying blocks.




Finally, the checkpointing based on Merkle Mountain Range (MMR), similar to FlyClient. The epoch-based syncing and BLS signature aggregation alone does not make the light client gas-efficient for cross-chain application. The BLS signature verification performs elliptic curve pairing causing 400K gas on Ethereum per verification. Harmony light client adopts a novel approach of MMR-based checkpointing for skipping frequent expensive pairing based verification.

Within every epoch, all blocks are connected to each other in the form of a MMR and the MMR root is included in every block header. By verifying the MMR commitment, it become possible for light client to validate the block,

**Peter Robinson** @drinkcoffee2010

Crosschain Workshop 2021: Agenda Announced! March 9 and 10 (or 8 and 9 depending on your timezone). The workshop aims to bring together world leading researchers and application developers to talk about cross-blockchain communications.
crosschain.mx/workshop2021/a...

1:26 PM · Feb 27, 2021

 142  16  Copy link to Tweet