# The Hemi Network

A modular protocol for superior scaling, security, and interoperability with Bitcoin and Ethereum

August 29, 2023

v1.6

**hemi.xyz**

---

# Introduction

The Hemi Network ("Hemi") is a modular protocol that integrates Bitcoin[1] and Ethereum[2] in a way that amplifies and extends the core capabilities of the two leading blockchain networks.

Hemi represents a novel perspective on how to address blockchain interoperability and scaling by approaching Bitcoin and Ethereum as components of a single supernetwork rather than two disparate ecosystems. This approach aims to:

- Harmonize these leading networks into a secure, scalable, and resilient protocol.
- Maximize the utility of the immense value stored across Bitcoin and Ethereum.
- Provide the foundation for further integrating the best features of blockchain technology with the broader Internet.

The core of the system is the Hemi Virtual Machine (hVM). The hVM envelops a full Bitcoin node within the Ethereum Virtual Machine (EVM). The Hemi Bitcoin Kit (hBK) makes hVM's new Bitcoin interoperability features accessible for developers. Hemi is thus designed to be as familiar to developers as an Ethereum Layer-2 network while fully absorbing and surfacing Bitcoin's capabilities. Applications that take advantage of Hemi's Bitcoin awareness or dual-network asset system are termed "hApps" to denote their multi-chain capabilities.

The benefits of the Hemi approach include the following:

- **Proof-of-Proof** [3] **Superfinality: —** Transactions on Hemi achieve Bitcoin-level finality in just a few hours, allowing Sequencer decentralization without sacrificing Ethereum settlement speed.
- **Tunnels: Trustless and Trust-Minimized Cross-chain Portability —** Since the hVM maintains protocol-level awareness of Bitcoin's and Ethereum's respective states, it enables superior methods of securely moving assets across chains.
- **hVM and hBK: True Bitcoin DeFi —** Hemi provides smart contracts with highly granular indexed views of Bitcoin state, enabling trustless DeFi applications and interoperability infrastructure that were previously impossible to build on an EVM.
- **Chainbuilder: Instant Extensibility —** External project teams can launch Hemi ecosystem chains (hChains) that harness Hemi's Bitcoin-Security-as-a-Service (BSaaS) capabilities and dual-chain interoperability.

- **Encapsulation: Onchain Asset Programmability** — Hemi offers advanced asset handling capabilities such as on-chain routing, time-lock, password-protect, and more.

The outcome is a network that not only provides an ideal surface area for development on Bitcoin and Ethereum, but makes possible an entirely new ecosystem of multi-chain interoperability secured by Bitcoin.

# Discussion of Prior Approaches

The Hemi Network synthesizes the flexibility of Ethereum with the security of Bitcoin, all while extending the capabilities and usability of both.

In this context, it's worth briefly assessing the conventional approaches to scaling both protocols and integrating Bitcoin security into smart contract networks.

## Approaches to Bitcoin Interoperability

- **BTC Relay** [4] — In this method, a smart contract system incentivizes third parties to relay Bitcoin headers to the protocol, which validates header PoW solutions and constructs a lightweight view of the canonical BTC chain. Smart contracts using this interoperability technology are limited to validating the presence of particular Bitcoin transactions in the canonical chain and rely on external relayers for proper function.
- **BeL2** [5] —This is a method of proving the existence of Bitcoin transactions to smart contracts using zero-knowledge proofs instead of Merkle proofs, combined with a collateralized escrow primitive based on a 2-of-3 multisig between participants to facilitate certain types of transactions. This approach also relies on transaction relayers and only provides smart contracts with proof of inclusion of specific transactions in the canonical Bitcoin chain.
- **Chain-key ECDSA Bitcoin Integration** [6] — In this method, block validators work together to create a shared ECDSA key. They can change who is involved by resharing the key. Smart contracts can manage Bitcoin wallets controlled by this key. Outgoing transactions from each wallet are signed by the key owners when requested by the managing smart contract. This method is limited to the validator set and the design of this single system. Block validators handle Bitcoin queries

from smart contracts and include the results in the blocks they produce. Only UTXO data and BTC fee levels are visible to smart contracts.

## Approaches to Scaling Ethereum

To date, increasing scalability on Ethereum has involved several different approaches:

- **Optimistic Rollups** [7,18] — A method by which the protocol publishes transactions in batches to an L1, but performs transaction execution off-chain and commits the resulting state back to the L1. These commitments are assumed valid unless contested within a seven-day-long challenge period before permitting settlement to the L1. This introduces significant delays and puts the onus of fraud detection on the user. Additionally, today's optimistic rollups rely on centralized sequencers with opaque mempools (the queue of pending transactions) and centralized proposers to communicate chain state back to the L1.
- **Zero-knowledge (zk) Rollups** [8] — This method also bundles and validates transactions before committing them to their companion L1 chain, but posts an irrefutable cryptographic proof to the L1 that the execution state is correct. As with optimistic rollups, today's zk-rollups also rely on centralized sequencers with opaque mempools and centralized proposers.
- **Validiums** [9] — This method is similar to a zk rollup, except that only state roots are published to the L1, and the network itself is responsible for making sure chain data is publicly available to users to facilitate settlement back to the L1.
- **Sidechains** [10] — This method involves a separate, independent blockchain connected to the main chain via a two-way bridge. Sidechains maintain their own independent consensus and cross-chain transactions rely on the bridge to ensure asset security.

## Approaches to Scaling Bitcoin

Similarly, a number of simpler approaches have also been tried to scale Bitcoin's throughput and feature set:

- **Federated Peg Sidechains** [11] — This method involves a separate blockchain network with a federated or centralized bridge mechanism to move funds to and from the sidechain. These sidechains can add functionality like Turing-complete smart contracts and zk-based privacy, but federated bridge operators must be trusted not to steal funds.

- **Drivechains (BIP 300)** [12] — This flavor of sidechain uses a bridge mechanism governed by Bitcoin miners, who could collude to steal bridged assets. Deploying this solution would require adoption of BIP 300 by the Bitcoin network, and withdrawing assets from a Drivechain back to Bitcoin would take three months.
- **Lightning** [13] — This method involves creating state channels on Bitcoin, allowing for rapid movement of funds between participants off-chain. Bitcoin deposited into Lightning channels is encumbered and can be trapped for months if a participant is malicious. Furthermore, Lightning channels are not capable of advanced programmability. The "hub-and-spokes" model of the current Lightning Network increases liquidity and efficiency, but at the cost of centralization and control over payment routes through the network.

## Approaches to Inheriting Bitcoin Security

Past approaches to integrating Bitcoin security into other networks generally fall into four categories:

- **Merged Mining** [14] — This approach encourages Bitcoin miners to mine blocks on another chain in parallel with Bitcoin. Merged mining not only requires the active participation of Bitcoin miners, but these Bitcoin miners can collude to attack the new chain at zero cost while continuing to mine Bitcoin legitimately. Merged mining is often used as the consensus protocol for sidechains.
- **Blind Merged Mining (BIP 301)** [15] — This approach iterates on merged mining by introducing a marketplace for users of various sidechains to construct sidechain blocks and bid for Bitcoin miners to merge-mine them. While this removes the need for Bitcoin miners themselves to run full nodes of all merge-mined chains, it creates a marketplace where anyone can purchase 51% attacks against any merge-mined chain for a nominal net cost.
- **Meta-Protocol** [16] — This approach seeks to deliver additional functionality by embedding transactions for new protocols directly in the Bitcoin blockchain itself. Bitcoin's block size and transaction fees, however, severely limit this approach even with mild adoption (e.g., Ordinals, BRC-20 tokens). Additionally, consensus on most feature changes in the core Bitcoin protocol to support new use cases has historically been nearly impossible to achieve, significantly limiting the functionality of these on-protocol additions.
- **Proof-of-Transfer (PoX)** [17] — Here, miners anchor a blockchain to Bitcoin by having the miners send BTC to stakers for the opportunity to mine a block. This system tightly couples block production and Bitcoin security inheritance, preventing the use of alternate consensus protocols and allowing the highest bidder to control block production.

Described in detail in the section to follow, the Proof-of-Proof (PoP) consensus mechanism employed by Hemi improves upon these previous attempts in the following ways:

- Bitcoin miners will secure Hemi and earn transaction fees without having to actively participate in — or even be *aware* of — the latter network's consensus or any layer on top of it.
- Transaction throughput in the Hemi ecosystem scales without increasing its Bitcoin footprint and security costs, because PoP transactions from Chainbuilder-supported chains are aggregated in Hemi's state roots published to Bitcoin.
- To perform a deep reorg, Bitcoin itself must also be 51% attacked. This is economically infeasible, even for nation-states.
- Block production is decoupled from Bitcoin security inheritance, allowing Hemi to utilize a consensus protocol which maximizes decentralization and long-term incentive alignment of sequencers.

By combining Bitcoin's PoW with Hemi's native decentralized sequencing, transactions on Hemi achieve greater security when compared to equivalently timed transactions on Bitcoin itself.

# Key Concepts

## Proof-of-Proof and Superfinality

Proof-of-Proof is a complementary consensus protocol that allows the Hemi Network to inherit Bitcoin's full Proof-of-Work security in a fully decentralized, trustless, transparent, and permissionless manner.

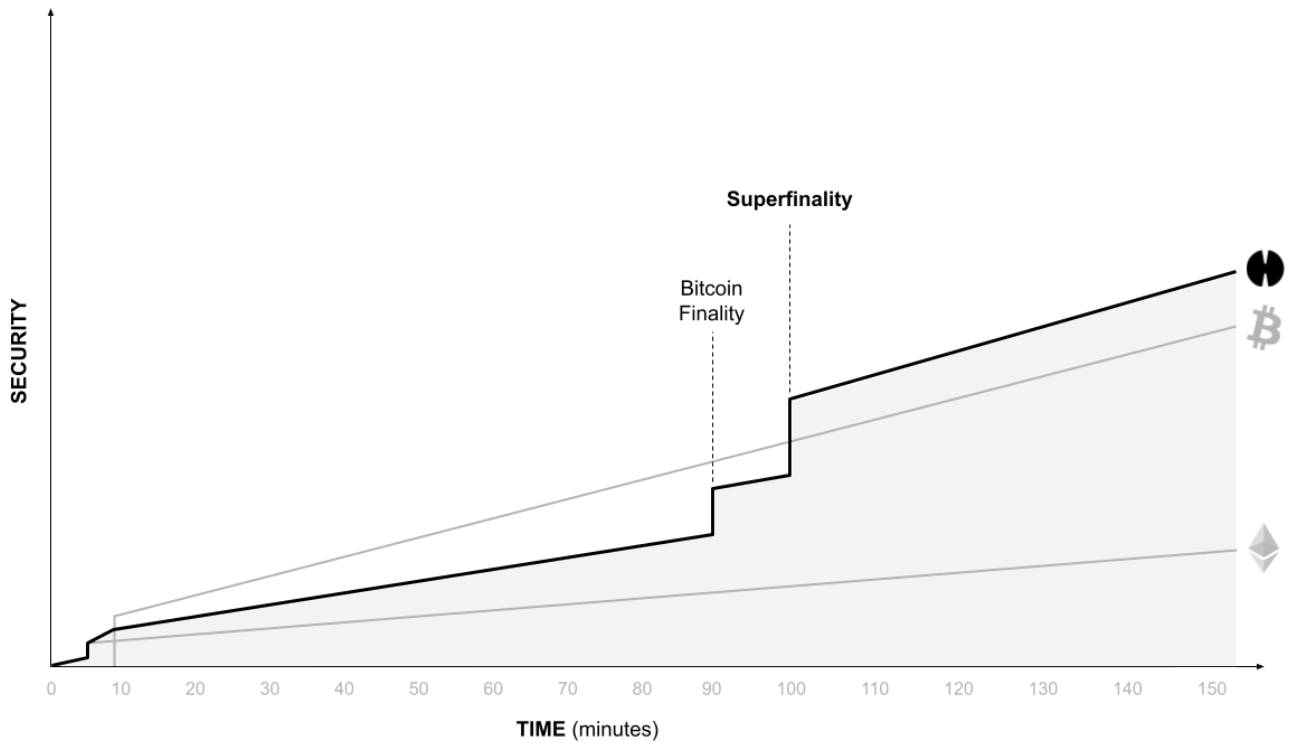
To inherit Bitcoin security, a new type of lightweight miner (a "Pop Miner") publishes Hemi consensus information to the Bitcoin blockchain.

After a state publication to Bitcoin, the Hemi protocol detects these publications through its protocol-level Bitcoin state awareness, with successful Pop Miners receiving a reward in the protocol's native token. The network uses these publications during fork resolution to prevent reorganizations with the full force of Bitcoin's security.

As Hemi's chain segments are PoP mined in the absence of publication of competing segments to Bitcoin, they receive Bitcoin confirmations. As a chain segment receives more Bitcoin confirmations, an attacker would have to control increasingly large ratios of staking power to create an alternate chain quickly enough to catch up in PoP publication weight and affect a reorganization.

During normal operation, in the absence of competing chain publications, each block on Hemi reaches full "Bitcoin finality" after nine Bitcoin blocks, or ninety minutes on average. At this point, it is mathematically impossible for anyone to reorganize the network without 51% attacking Bitcoin itself to retroactively insert PoP publications of an attacking chain.

This means that the protocol can mathematically prove the impossibility of reorganizing a particular Hemi block based on publicly available data on the Bitcoin blockchain. Further, any long-range attack must be publicly announced via Bitcoin transactions in lock-step with the block production of the challenged part of the legitimate chain. This provides advanced warning and makes the attack useless against any service or cross-chain hApp waiting for Bitcoin finality even if the reorganization is successful.

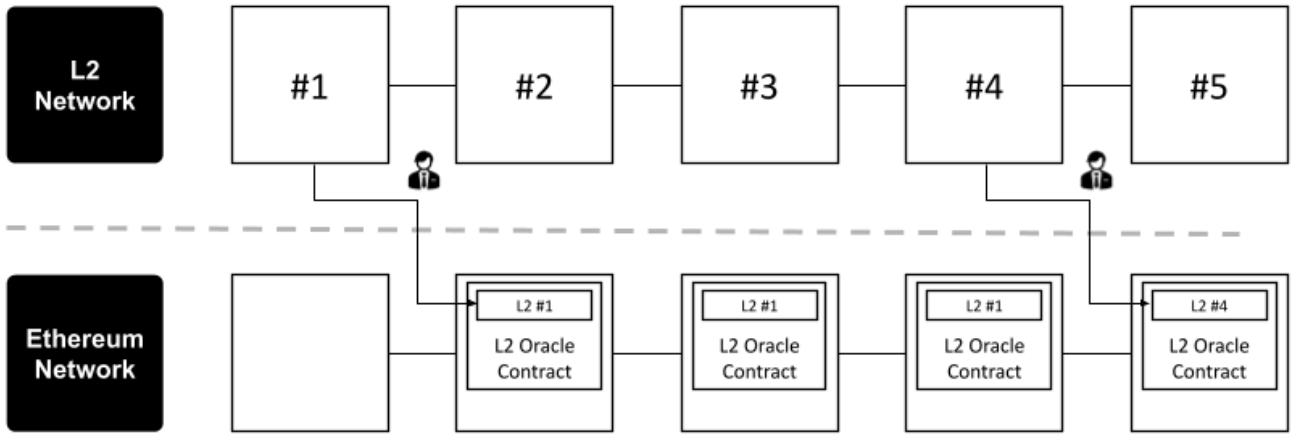


*This chart shows the general shape of the relative levels of security — that is, the cost to execute a 51% attack — for Bitcoin, Ethereum, and the Hemi Network over approximately three hours.*

Due to the Bitcoin finality delay of **nine** blocks, the security of a block at the moment of achieving Bitcoin finality is lower than that of a block produced on the Bitcoin network at the same time. However, due to its hybrid consensus protocol, a Hemi block reaches Superfinality after an additional Bitcoin block occurs, because an attacker would have to control majority consensus power on Bitcoin *and* Hemi simultaneously to have any reasonable statistical probability of affecting a reorg.

## Decentralized Rollup Mechanics

All L2 rollup chains must periodically publish their state roots back to their companion L1 to allow asset bridging and other cross-chain smart contract calls back to the L1. Today's L2 networks employ a centralized "proposer" who pays the Ethereum gas fees essential to this critical process.




*In conventional L2 deployments, centralized Proposers periodically publish L2 state back to Ethereum for settlement.*

Even with a decentralized sequencing protocol, a centralized proposer could cause the entire L2 to halt by refusing to publish valid rollup state roots back to Ethereum. This halting power could also be used to coerce decentralized sequencers into exerting censorship by refusing to communicate state rollups of any chain segments containing proposer-blacklisted transactions back to Ethereum for settlement.

Invalid publications on optimistic rollup networks can lead to asset theft by faking L2 states. To prevent this, rollups use actors to challenge these invalid states with fault proofs through an interactive challenge-response protocol. This process requires a lot of ETH for gas. Recently, some optimistic rollups

have introduced a bond system where a proposer must post collateral when making a state claim. To challenge the claim, an actor must also post a bond. As the fault dispute game progresses, the participants must post increasingly large bonds, which the prevailing honest party will receive upon resolution of the game.

In this system, the initial bond for each state claim is small, which offers minimal incentives for decentralized participants to actively monitor state roots. A successful challenge where the dishonest proposer instantly abandons the game will only pay out the initial state root publication bond, which in practice could be a few hundred dollars.

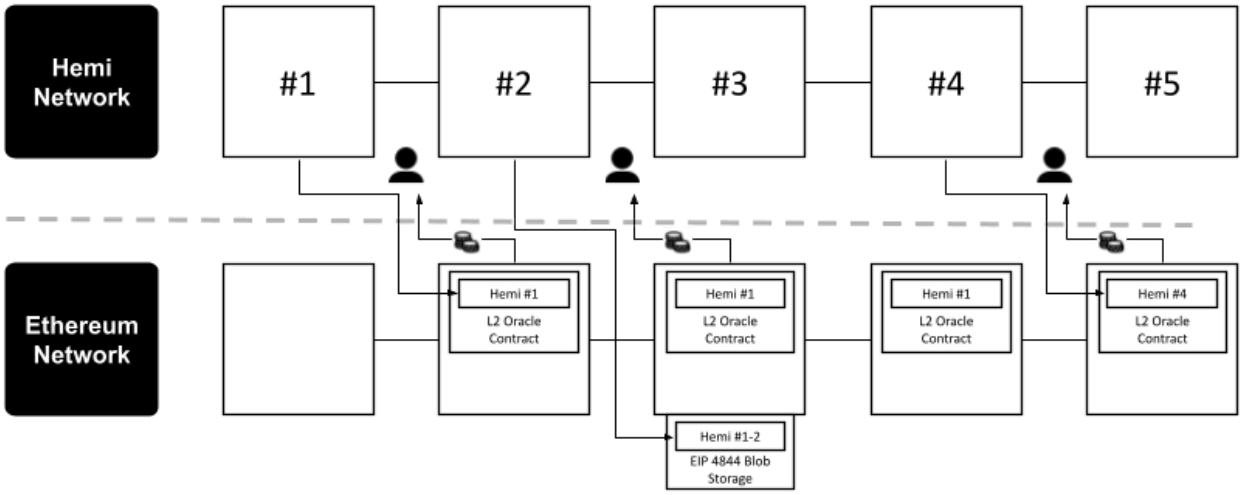
Today's L2 networks also employ a centralized "batcher" to compress and publish batches of L2 transactions back to Ethereum for data availability ("DA") which ensures the canonical L2 can be derived entirely from the L1. A centralized batcher could perform denial-of-service attacks by refusing to publish batches back to Ethereum, ultimately resulting in the L2 experiencing a large reorganization due to the protocol rederiving an empty chain in the absence of batch publications within the timeout window.

### Decentralized Publication and Validation

As detailed in "System Design Overview," anyone can participate in the incentivized process of publishing Hemi data to Ethereum by staking sufficient native tokens.

Similar to Pop Miners, who receive rewards for publishing Hemi state data to Bitcoin, Publishers compete to publish data to Ethereum in exchange for rewards paid out in the protocol's native token.

Hemi Publishers perform the combined roles of "proposers" and "batchers" in other L2 networks. They perform each role asynchronously to prevent either role from bottlenecking the other.

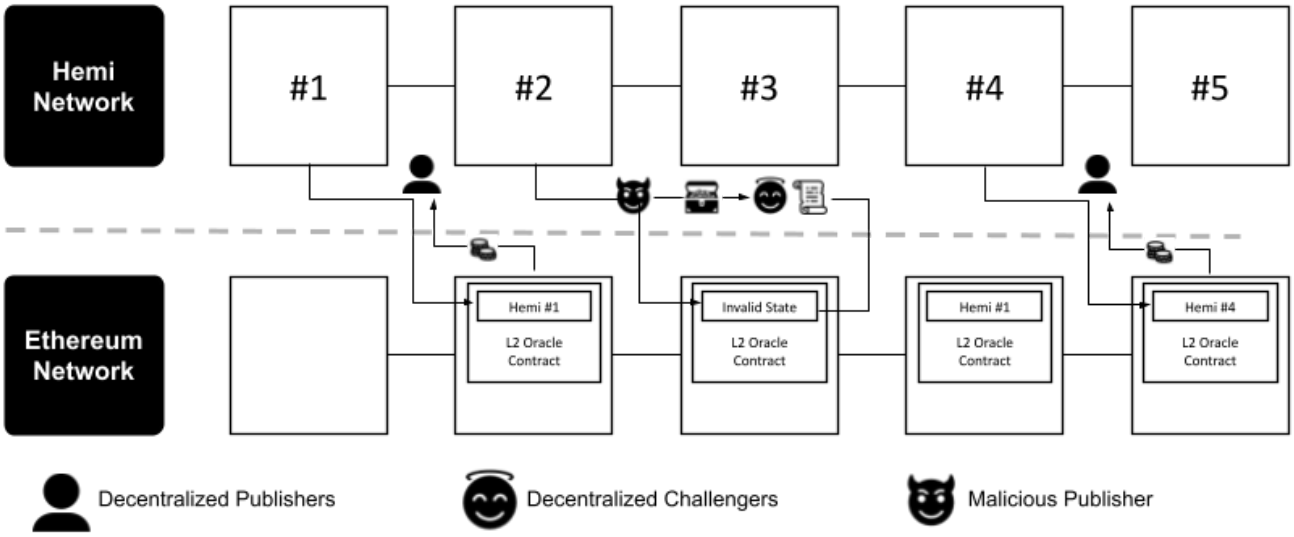



*Collateralized Decentralized Publishers on the Hemi Network periodically publish state roots for settlement and block batches for data availability to Ethereum and collect a reward paid by the Hemi protocol.*

Hemi validation contracts manage the rewards given to Publishers to make sure they are adequately incentivized to publish data during fee market spikes.

Additionally, Hemi introduces decentralized "Challengers" who monitor Publisher activity and challenge invalid publications with fault proofs. When a Challenger successfully proves a publication is invalid, the misbehaving Publisher has their stake slashed. During the slashing, a part of the misbehaving Publisher's stake is burned by the protocol and the remainder is awarded to the Challenger who submitted the fault proof.

This system of staking a single larger quantity of tokens once rather than posting small bonds for each publication provides a stronger deterrent against misbehavior, and a larger incentive for users to run verification nodes which proactively scan for invalid publications to challenge.




*When a malicious Publisher submits an invalid state root to Ethereum, anyone can participate as a decentralized Challenger and submit a fault proof, receiving a portion of the malicious Publisher's stake as a reward. Fault proofs require an interactive dispute game between Publisher and Challenger not shown in diagram.*

## Asset Portability: Tunnels

"Tunnels" describe how digital assets move between Hemi and the Ethereum and Bitcoin networks. Hemi's unique consensus-level knowledge of Bitcoin's and Ethereum's respective states enables a variety of methods for moving assets between networks beyond the traditional methods offered by conventional bridges.

The decentralized inheritance of Bitcoin's security ensures that all assets transferred to and from Bitcoin and Ethereum are protected against depeg and reorganization attacks and can even operate securely in the context of a malicious consensus supermajority.

### Bitcoin Tunnel

Moving tokens between EVM-compatible networks is well-understood. However, moving Bitcoin and other Bitcoin-native assets to an EVM and back without relying on a centralized party is less common. This is because an off-chain entity must hold the actual Bitcoin to back the synthetic assets on the EVM chain. Many of these "bridges" have been insecure and vulnerable to hacks.

Hemi's Bitcoin state knowledge enables a range of centralized and decentralized custodianship approaches to Bitcoin asset portability. These approaches can have different risk, speed, and cost tradeoffs. Paired with atomic swaps for immediate fungible asset redemption, this will enable a highly liquid and performant system for tunneling Bitcoin assets, including Ordinals and BRC-20 tokens, to Hemi and through to the broader Ethereum ecosystem.

Hemi will provide two different asset custodianship systems; low-value asset vaults secured by overcollateralized multisig, and high-value asset vaults secured by a variation of BitVM. The "Bitcoin Tunnel Implementation" section provides a detailed explanation of this dual-custodianship model. Both vault systems are built as smart contracts on hVM, and hApp developers can launch additional Bitcoin Tunnel systems as well.

### Ethereum Tunnel

Moving Ethereum assets to and from Hemi functions similarly to other optimistic rollups, except with faster settlement due to Bitcoin Finality and the decentralization of the Challenger role.

When a Hemi user wishes to deposit assets, the user initiates a deposit transaction on Ethereum which locks up the assets in the Hemi validation contracts on Ethereum. Once the deposit is incorporated into an Ethereum block, Hemi's block derivation protocol requires the Sequencer to include the deposit in the first Hemi block derived from the Ethereum block in question. This inclusion mints representative tokens for the user, which completes the Tunnel deposit.

To move assets back to Ethereum, a Hemi user sends a withdrawal transaction on Hemi, which burns the representative tokens. After a few minutes, a Publisher submits an updated rollup state root to Ethereum. Once this root is available, the user submits a withdrawal proof demonstrating that the representative tokens were appropriately burned. After Bitcoin finality is achieved and in the absence of any Challenger initiating the fault dispute process at or before the state root, the withdrawal is finalized, and the user claims the corresponding assets from the Hemi validation contracts.

Hemi also supports tunneling Hemi-native assets, including Bitcoin-native tunneled assets, out to Ethereum. To do this, a Hemi user sends a native asset deposit transaction on Hemi, locking up the assets in the Hemi native asset tunnel contract. Similar to withdrawing Ethereum assets, the user waits for a Publisher to submit an updated rollup state root to Ethereum and then submits a native asset deposit proof on Ethereum. After Bitcoin finality is achieved, and if no Challenger initiates the fault dispute

process, the user submits a native deposit claim transaction, prompting the Hemi validation contracts on Ethereum to mint tokens representing the Hemi-native assets.

Tunneled Bitcoin-native assets are considered Hemi-native because the representative token originates on Hemi. Hemi-native tokens are any ERC-20 token that originates on Hemi. The protocol also supports tunneled Ethereum assets, which are any ERC-20 contract on Hemi that represents an ERC-20 contract originally from Ethereum.

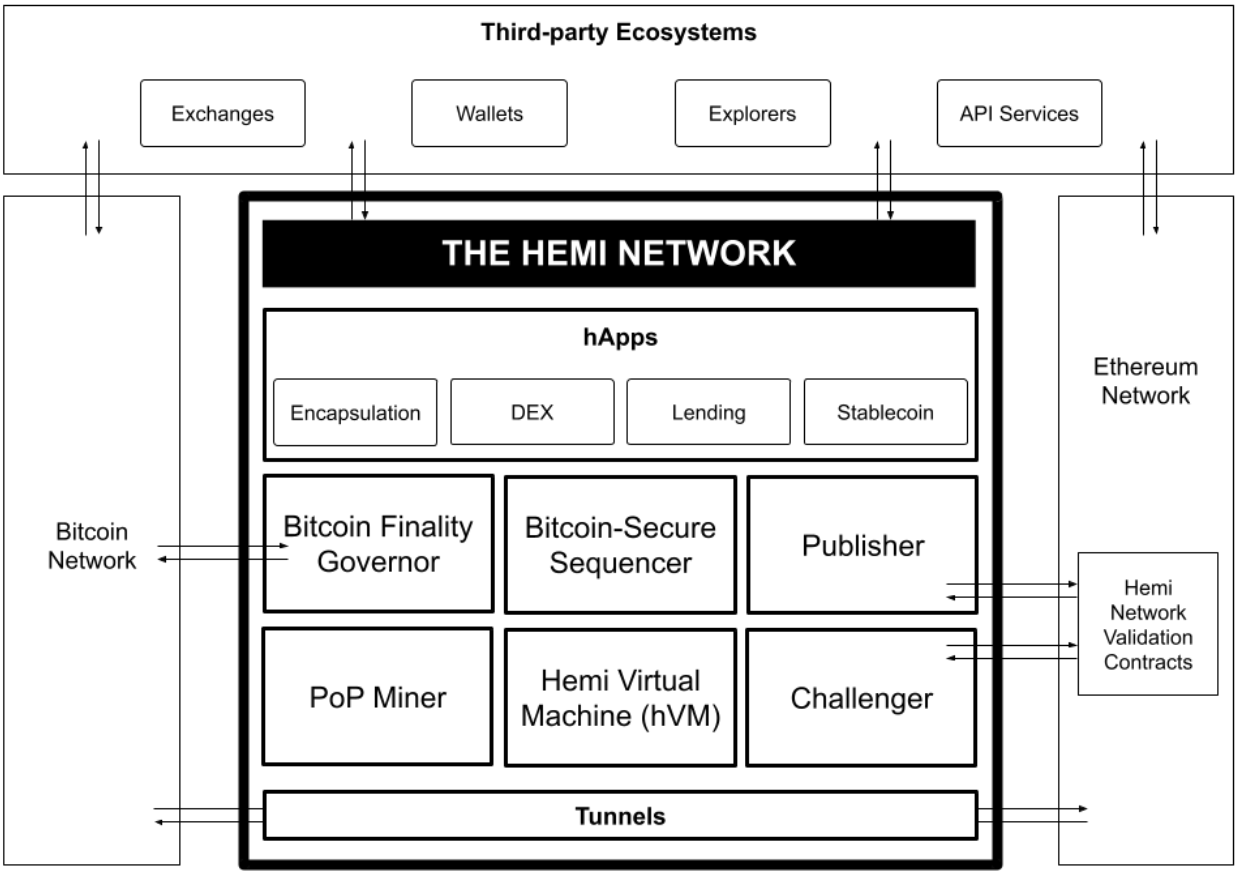
To transfer Hemi-native assets back to Hemi, the user initiates a native asset withdrawal transaction on Ethereum, burning the Ethereum tokens representing the Hemi-native assets. Once this transaction is included in an Ethereum block, Hemi's block derivation protocol requires the Sequencer to process the withdrawal transaction in the first Hemi block derived from the relevant Ethereum block. In this Hemi block, the withdrawal is processed, and the corresponding Hemi-native assets are transferred from the Hemi native asset tunnel contract to the user.

Depositing a Hemi-native asset to Ethereum involves a process similar to withdrawing an Ethereum-native asset back to Ethereum. The difference is that the final step on Ethereum involves minting a representative token for the Hemi-based asset. Similarly, withdrawing a Hemi-native asset from Ethereum follows a process similar to depositing an ETH-native asset, except the final step is releasing the original Hemi-native token rather than minting a representative token.

This system of tunneling Hemi-native assets to Ethereum is particularly useful because it allows Bitcoin-native assets to be tunneled through Hemi into the broader Ethereum ecosystem. As a result, dApps on Ethereum or any Ethereum L2 can utilize Bitcoin-native assets protected by Hemi's tunnel system.

# System Design Overview: The Hemi Network

The general architecture of the Hemi Network is as follows.




As a system, Hemi's components manage:

1. Decentralized blockchain progression
    a. Performing block derivation from Ethereum
    b. Incorporating  new Bitcoin block headers to synchronize hVM state progression
    c. Calculating and executing PoP payouts
    d. Transaction gathering from mempool and sequencing
    e. Publication of Hemi block batches to Ethereum for DA
2. Inheritance of Bitcoin PoW security
    a. Decentralized Hemi state publication to Bitcoin via PoP transactions
    b. Executing PoP fork resolution
    c. Processing of Bitcoin blocks and mempool to provide Bitcoin finality information
3. Providing EVM-level Bitcoin state awareness through hVM
    a. Progressing hVM's Bitcoin view
    b. Exposing the Bitcoin view to the EVM through precompile contracts

4. Tunneling of Ethereum and Hemi assets
    a. Extracting and processing Ethereum deposit and Hemi withdrawal transactions during block derivation
    b. Decentralized publication of Hemi data to Ethereum
    c. Managing fault proofs to ensure the integrity of data
    d. Verifying withdrawal and deposit proofs on Ethereum
5. Tunneling of Bitcoin assets using hVM
    a. Constructing and maintaining both types of Bitcoin asset vaults
    b. Detecting and processing asset deposits
    c. Coordinating vault operators to facilitate withdrawals
    d. Detecting and punishing vault operator misbehavior

All incentivized network participants are rewarded for performing their roles via native tokens. Participants in roles that require payment of BTC/ETH fees supply these assets themselves, and the protocol economics are engineered to reward participants sufficiently to cover the cost of acquiring these assets.

Hemi supports a dual-asset gas model which allows users to pay fees in ETH or native tokens. To maximize compatibility and ease of use as an Ethereum rollup, Hemi defaults to ETH gas payments which the protocol automatically converts to native tokens using on-chain DEXes before distributing to the appropriate parties. ETH gas payments are charged a conversion fee, incentivizing users to pay fees in native tokens without introducing adoption hurdles.

| Actor | Role | Network Token Staker | Resources Used by Actor | Incentive Token Source |
|---|---|---|---|---|
| Sequencer | Sequencing (PoS mining) | Yes | Running nodes | Protocol emissions + Tx fees |
| Pop Miner | PoP mining (Publishing Hemi state to Bitcoin) | No | BTC for Bitcoin network fees | Protocol emissions + Tx fees |
| Publisher | Proposing (Publishing Hemi consensus state to Ethereum) | Yes | ETH for Ethereum gas fees and running nodes | Protocol emissions + Tx fees |
| | Batching (Publishing Hemi block contents to Ethereum) | | | Tx data availability fees |
| Challenger | Challenging (Sending | No | ETH for Ethereum gas | Slashed stake from |

| Actor | Role | Network Token Staker | Resources Used by Actor | Incentive Token Source |
|---|---|---|---|---|
| | fault proofs when Publishers misbehave) | | fees and running nodes | misbehaving Publisher |

† Details on the Hemi Network's governance model to be detailed in a separate document.

Details of each component's operation follow.

## Pop Miner

Users who want to participate in securing Hemi to Bitcoin and earn Hemi mining rewards from the protocol run the Pop Miner. The application:

1. fetches Hemi headers from the Bitcoin Finality Governor for publication to the Bitcoin blockchain;
2. constructs Bitcoin transactions containing encoded Hemi headers; and
3. sends these Bitcoin transactions to the Bitcoin Finality Governor for broadcasting.

## Bitcoin Finality Governor (BFG)

Bitcoin finality refers to the Bitcoin-level security that transactions on Hemi achieve through the processing stage. The BFG provides these Bitcoin finality security statistics, and facilitates Bitcoin network access for other components of the Hemi stack.

The BFG daemon:

1. relays Hemi block headers to Pop Miners;
2. facilitates Pop Miner communication with the Bitcoin network to retrieve spendable UTXOs and propagate PoP transactions containing Hemi state publications;
3. parses full Bitcoin blocks and the Bitcoin mempool for the presence of PoP transactions containing Hemi headers; and
4. processes Hemi headers embedded in PoP transactions to determine Bitcoin finality status of Hemi chain segments, detecting the presence of any attacking chain publications.

If no attacks are mathematically possible for a given chain segment without Bitcoin itself being attacked, BFG marks the chain segment as having achieved Bitcoin finality. However, if a potential reorganization appears on the Bitcoin blockchain within the Bitcoin finality period, Bitcoin finality (and Superfinality) is delayed for the affected chain segments until the attack stops and PoP publications of a single chain prevail without competing publications. In either case, third-party applications like exchanges and wallets use BFG to analyze transaction security in real-time and dynamically confirm transactions upon reaching Bitcoin finality.

## Hemi Virtual Machine (hVM)

The hVM is an EVM enhanced with Bitcoin interoperability and presents as an EVM wrapped around a full-indexed Bitcoin node. The processed Bitcoin data exposed by hVM to smart contracts enables developers to build decentralized and trustless versions of Bitcoin DeFi applications in a more secure, cost-effective, and simple manner. Such applications would otherwise require expensive proof validation and trusted oracles or impractical fault-proof systems for processed data that can't be cryptographically authenticated.

It processes smart contract execution, maintains EVM state, and facilitates peer-to-peer communication of pending transactions.

The hVM:

1. manages the fetching of full Bitcoin blocks to perform synchronized progression of hVM Bitcoin state;
2. processes transactions contained in execution-layer payloads provided by the BSS;
3. facilitates smart contract access to the embedded Bitcoin full node through new precompile contracts;
4. extracts PoP transactions from its embedded Bitcoin node for the calculation and execution of PoP payouts; and
5. manages the Hemi mempool by receiving, validating and propagating pending transactions, and providing pending transactions to BSS for block creation.

## Hemi Bitcoin Kit (hBK)

The Hemi Bitcoin Kit (hBK) is a set of smart contracts that abstract away low-level interaction with the Bitcoin interoperability precompiles available in the hVM.

The hBK:

1. translates smart contract queries for Bitcoin data into precompile calls;
2. parses the raw data returned by the precompiles into structures that make the retrieved Bitcoin data easy to use; and
3. provides endpoints that perform multiple precompile calls and synthesizes the results to answer more complex queries efficiently.

Additionally, hBK provides integrations with popular development tools which make these smart contracts easy to leverage while building Bitcoin-aware hApps.

## Bitcoin-Secure Sequencer (BSS)

The Bitcoin-Secure Sequencer (BSS) manages and maintains Hemi consensus and handles bidirectional communication between Hemi and Ethereum. It consists of two daemon processes.

The BSS daemons:

1. handle Sequencer staking/unstaking/slashing operations;
2. communicate with the hVM to gather mempool transactions and Bitcoin headers to construct Hemi blocks using rollup block derivation from Ethereum blocks;
3. perform block validation and fork resolution to maintain Hemi's consensus layer;
4. derive the Hemi chain from batches published to Ethereum; and
5. provide the Publisher and Challenger components with Hemi consensus state and Bitcoin security data.

## Publisher

Publishers communicate Hemi transaction and consensus data to Ethereum to provide DA and settlement. Users must stake Hemi's native token to operate as a Publisher and earn rewards.

## Challenger

Challengers make sure Publishers only commit full and correct state to the Hemi contracts on Ethereum. Challengers who correctly identify and prove incorrect state publications receive a part of the Publisher's stake.

## Hemi Validation Contracts on Ethereum

To track Hemi consensus and rollup state and to provide data availability for block content, several Ethereum contracts receive and process data from Publishers.

These contracts also receive fault proofs from Challengers in the event a Publisher submits incorrect or incomplete state roots and coordinates the interactive fault dispute game. Further, they will manage the slashing of misbehaving Publishers and the rewarding of Challengers for successful fault proofs.

## Tunnels

This component of Hemi facilitates cross-chain interoperability as described above. At the highest level, tunnels:

- manage centralized and decentralized custodianship systems and P2P cross-chain swaps for Bitcoin assets; and
- facilitate the movement of digital assets between Hemi and Ethereum.

The Bitcoin Tunnel:

1. manages the creation and maintenance of custodianship vaults;
2. verifies successful deposit of BTC and other Bitcoin-native assets to one of the Bitcoin tunnel custodianship systems;
3. mints tokens representing the deposited asset and sends them to the Hemi address of the depositor;
4. burns tokens representing deposited assets when a Hemi user initiates a tunnel transaction back to Bitcoin;
5. informs the relevant tunnel custodianship system of the successful initiation of a tunnel withdrawal; and

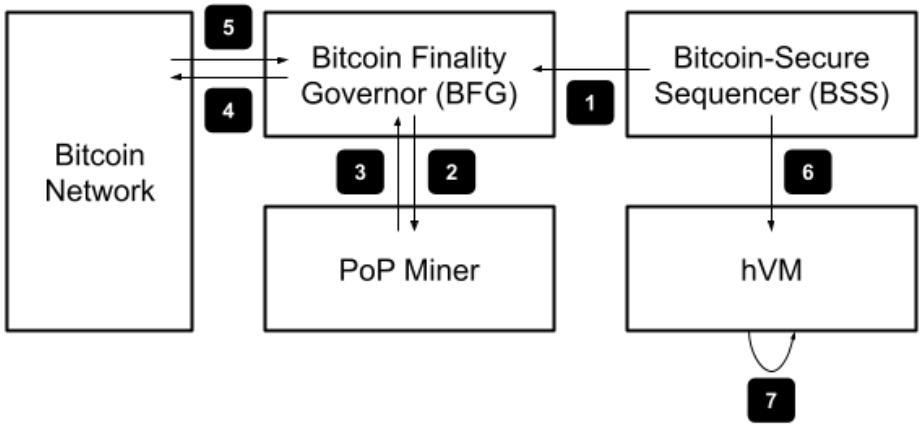
6. ensures the custodianship system completes the prescribed tunnel withdrawal, taking actions such as re-routing tunnel withdrawals and slashing custodianship system participants for any misbehavior.

The Ethereum Tunnel:

1. verifies Ethereum assets have been successfully locked in the tunnel contract on Ethereum;
2. mints tokens representing the locked assets from Ethereum and sends them to the Hemi address of the tunnel user;
3. burns tokens representing locked Ethereum assets and sends the corresponding assets to the Ethereum address of the depositor;
4. locks up Hemi-native and Bitcoin-tunneled assets for tunneling to Ethereum;
5. mints tokens representing the locked assets from Hemi on Ethereum and sends them to the Ethereum address of the tunnel user; and
6. burns tokens representing locked Hemi-native and Bitcoin-tunneled assets on Ethereum and sends the corresponding assets to the Hemi address of the tunnel user.

## PoP Mining Process Flow

This diagram shows how Hemi's components work together to perform decentralized inheritance of Bitcoin security:




1. A Bitcoin Secure Sequencer (BSS) produces a new block which is broadcast to other BSS nodes. BSS nodes send the new block header to Bitcoin Finality Governor (BFG) nodes.

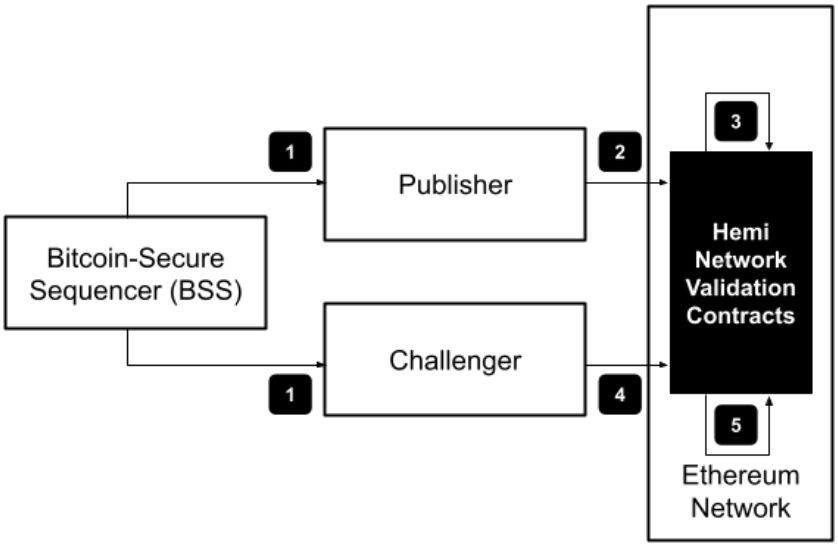
2. BFG nodes send the new block header to Pop Miners.
3. Pop Miners encode the new block header into signed Bitcoin PoP transactions which they return to BFG nodes.
4. BFG nodes propagate the Bitcoin PoP transactions to the Bitcoin network.
5. The Bitcoin PoP transactions are included in a Bitcoin block and BFG nodes detect this inclusion to update the Bitcoin finality security statistics.
6. A BSS node mines a new block containing new Bitcoin headers which communicate updated Bitcoin state to the hVM.
7. After the PoP mining window closes, hVM nodes retrieve all PoP transactions from their Bitcoin views, calculate the rewards owed to Pop Miners who secured the block to Bitcoin, and reward them with native tokens.

By the end of this process, a segment of the Hemi chain has been successfully secured to Bitcoin and participating Pop Miners have received their rewards.

## Hemi Validation Contracts Process Flow

For asset transfers and other cross-chain contract calls from Hemi to Ethereum to execute securely, the Hemi validation contracts running on Ethereum require updated information about Hemi consensus and canonical state.

This diagram shows how Hemi's components work together to perform decentralized publication and verification of Hemi state and security to the Ethereum-side contracts:




1. BSS nodes send Hemi consensus data to Publishers and Challengers.
2. Publishers compare the data provided by BSS nodes to data currently known by the Hemi Network validation contracts and publish any new information to the contracts through an Ethereum transaction.
3. Hemi Network validation contracts process new data from Publishers to update their view of Hemi state and security of its chain segments to Bitcoin.
4. Challengers compare the BSS data against data published to Hemi Network validation contracts by Publishers. If incorrect or incomplete data was published, Challengers submit a fault proof.
5. If a Challenger submits a fault proof, the Hemi Network validation contracts coordinate an interactive challenge-response process, fix the incorrect or incomplete information, slash the responsible Publisher, and deliver some of the Publisher's stake to the Challenger.

At the end of this process, the Hemi Network validation contracts on Ethereum have a complete view of Hemi's current state and the Bitcoin finality of its chain segments. Once a chain segment reaches Bitcoin finality and cannot be reversed without a 51% attack on Bitcoin, the Hemi Network validation contracts allow cross-chain withdrawal transactions from that chain segment to complete execution and settle on Ethereum.

# Implementation of Bitcoin Interoperability and Security

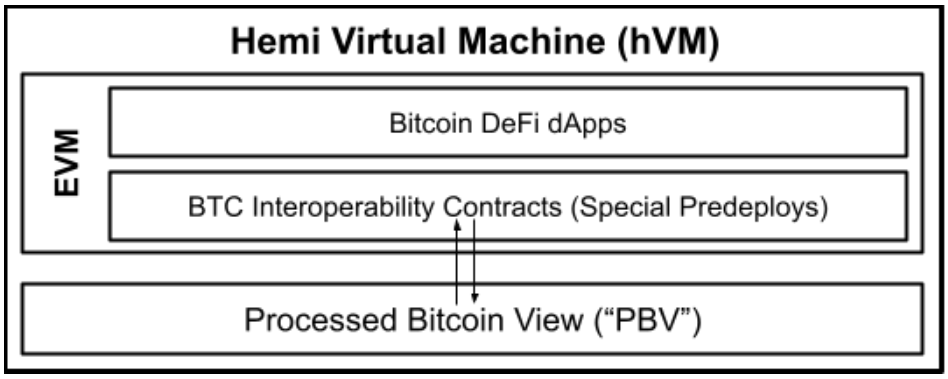
## Bitcoin Interoperability: hVM

The Hemi Network introduces a new approach to building Bitcoin-aware applications – maintaining a deterministic "Processed Bitcoin View" (PBV) via the hVM described above.

The PBV tracks traditional Bitcoin state (including the UTXO table and address history), calculates transaction and block metadata statistics and operates additional indexers with awareness of Bitcoin-based meta-protocols like Ordinals/Runes/BRC-20.

The hVM also provides a novel Bitcoin event subscription and notification subsystem that allows smart contracts to automatically receive callbacks when Bitcoin events occur, rather than relying on traditional fault-prone, complex, and expensive solutions based on third-party agents relaying data or triggering actions through externally owned accounts (EOAs).

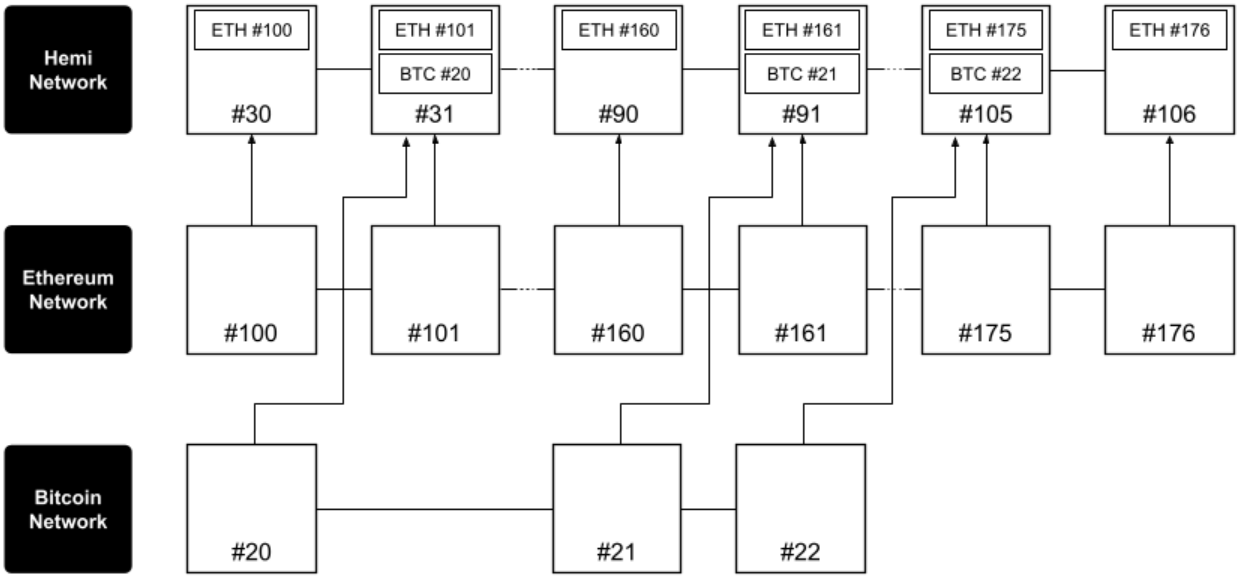
By combining robust EVM-level access to Bitcoin's state with automatic smart contract callbacks on relevant Bitcoin events, Hemi enables developers to build sophisticated hApps and infrastructure in a simple, gas-efficient, and secure manner not possible with conventional Bitcoin interoperability solutions.




*DeFi apps query new Bitcoin interoperability predeploys which communicate with the PBV.*

## Maintaining a Deterministic Bitcoin View

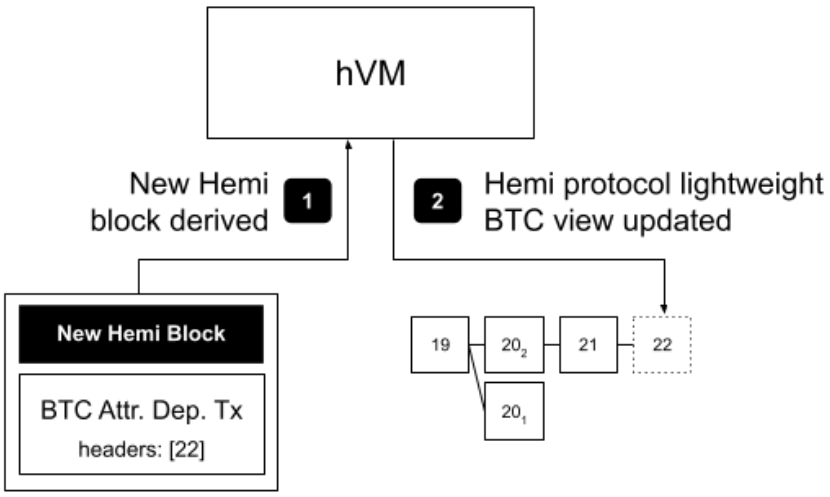
For Hemi's hVM to serve a deterministic view of Bitcoin to smart contracts, Hemi incorporates Bitcoin blocks into the L2 derivation process alongside Ethereum blocks. New Bitcoin block headers are incorporated into the Hemi chain through a new "BTC Attributes Deposited" transaction. This transaction is rolled up to Ethereum.




*Some Hemi blocks are also derived from Bitcoin blocks, deterministically updating Hemi's Bitcoin view.*

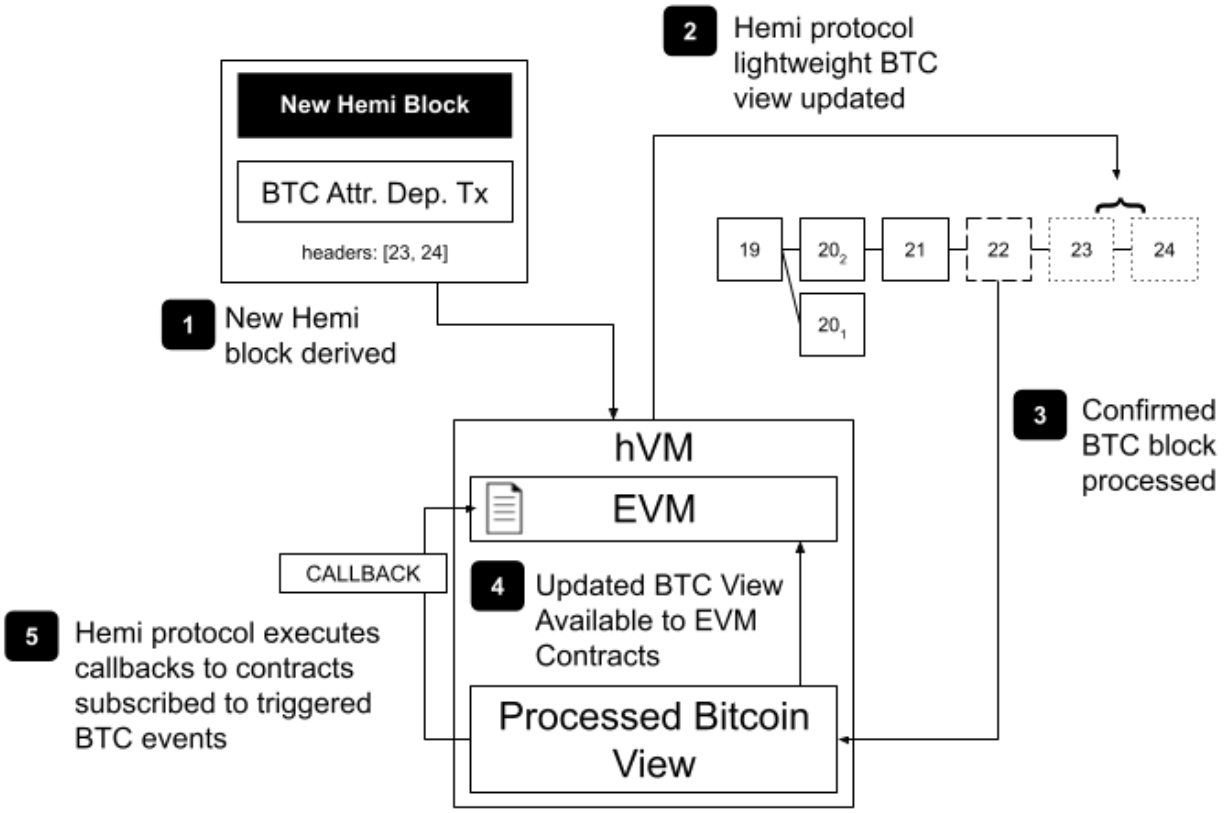
When a new Bitcoin block occurs, a Sequencer can elect to incorporate its header into the L2 block the Sequencer is responsible for building using the special BTC Attributes Deposited transaction, making the Hemi protocol aware of the new BTC block. Sequencers will be incentivized to include Bitcoin headers

with an additional block subsidy. If the BTC block header is internally valid and connects to the hVM's lightweight BTC view without violating any BTC protocol rules, it is added to the lightweight view. Otherwise, the Hemi block is marked invalid and ignored.




*When a Hemi block is derived from a Bitcoin block, Hemi updates its lightweight canonical Bitcoin view.*

Then, the Hemi protocol waits for additional Bitcoin blocks to be communicated. Once a Bitcoin block communicated to the protocol receives sufficient BTC confirmations in the lightweight view, it is fetched from the Bitcoin network and processed by the EVM-level Bitcoin full node and indexers, after which the updated PBV is available to smart contracts. This confirmation delay protects Hemi against malicious data-withholding attacks by Bitcoin miners. When PBV processes the new Bitcoin blocks, it tracks whether the state updates trigger any smart contract's Bitcoin event subscriptions. For any subscription that is triggered, PBV generates and executes a Hemi transaction that calls the subscribed smart contract with the details of the event that occurred.



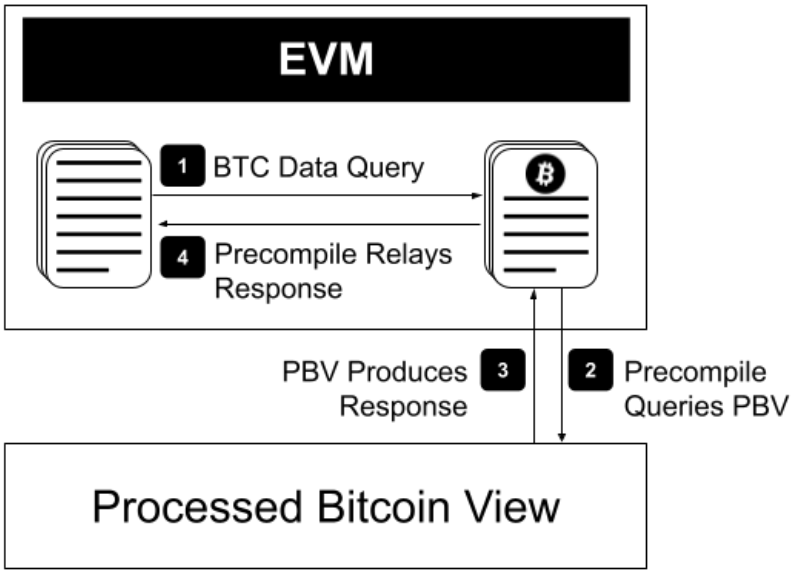

*Once a particular Bitcoin block reaches a sufficient number of confirmations in Hemi's lightweight view, it is processed in full by the PBV, making updated Bitcoin state available to hApps.*

This approach ensures that all Hemi nodes process the same Bitcoin block at the same Hemi block height. Thus, all EVM calls to PBV in a particular Hemi block, along with all the event-triggered callbacks, are deterministic based on Hemi state.

## Bitcoin State Smart Contract Queries

The hVM makes the deterministic PBV available to smart contracts using special Bitcoin Interoperability predeploy contracts in the EVM. These predeploy contracts expose a number of functions which represent queries against PBV.

When one of these predeploys is called, the appropriate query is performed against the PBV and the result is returned into the EVM, charging a fixed gas cost similar to Ethereum's existing precompiles.




*The Bitcoin interoperability predeploys provide hApps access to Bitcoin data in PBV.*

From the smart contract's perspective, this behaves identically to a full Bitcoin node and additional indexer services running inside the EVM. Updates to the Hemi protocol can add additional endpoints and new indexers for new Bitcoin metaprotocols.

The precompiles provide queries that let smart contracts access the following:

- UTXOs, balances, and transaction history by address/scripthash
- Confirmations of specific transactions
- Bitcoin tip and specific headers
- Transaction fee statistics
- Transaction position in blocks
- Transaction history and current owner of specific Ordinals Theory sats and associated inscriptions
- History of BRC-20/Rune transactions by address/script
- Distribution of specific BRC-20/Rune tokens

To make Bitcoin-aware hApp development easier, the hVM also includes the hBK described above — a collection of smart contracts that abstract away the complexities of direct precompile interactions. These smart contracts provide data structures, handle data marshaling/unmarshaling for precompile calls, and surface endpoints for higher-level queries that require multiple PBV calls or additional processing.

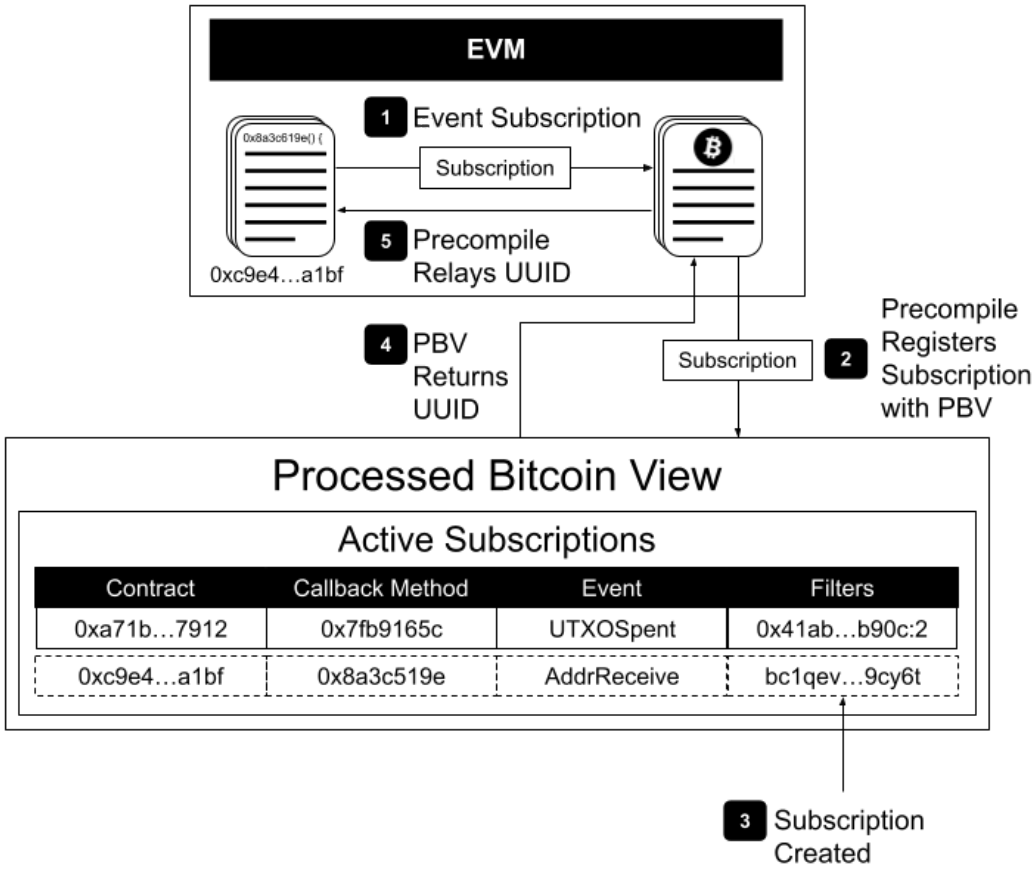
Bitcoin data returned by queries to the hBK are fully parsed and come with relevant metadata as appropriate, such as a transaction structure with all inputs, outputs, and metadata, or a Bitcoin block header structure that includes header fields along with height and cumulative chain difficulty.

### Bitcoin Event Notifications

The Hemi protocol also allows smart contracts to subscribe to Bitcoin event notifications, where the protocol automatically generates and processes callback transactions to smart contracts when certain events occur on Bitcoin.

This allows Apps to respond to any events on Bitcoin that are relevant to their protocol without requiring the application to incentivize fault-prone third-party participants to monitor Bitcoin on their behalf and manually trigger smart contract calls from an EOA.

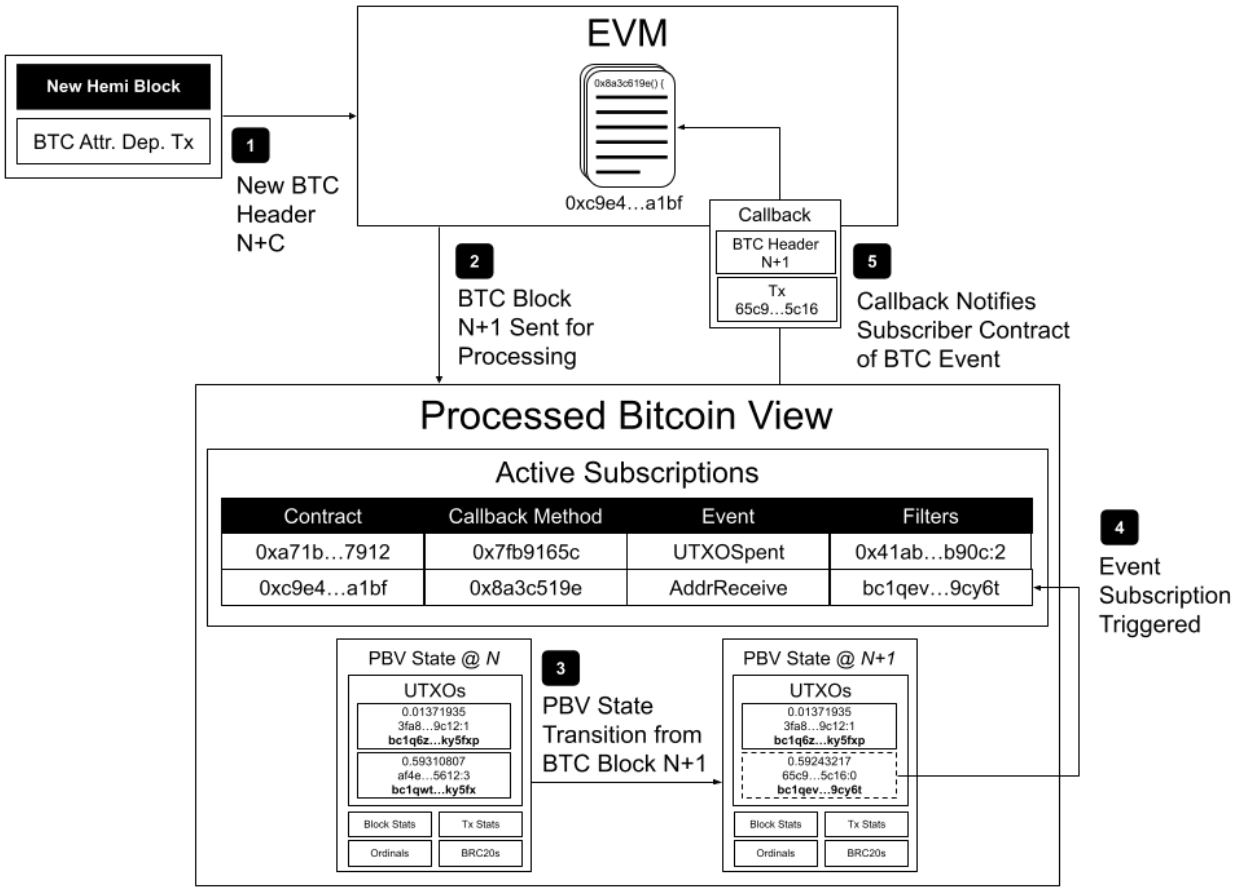
When a smart contract is running on the hVM, it can call a subscription Bitcoin interoperability predeploy, specifying which type of event the smart contract is interested in, what filters to apply, and the method in which the smart contract should receive the callback when the event occurs. The subscription call returns a UUID for the subscription, which the smart contract can use to modify or cancel the subscription in the future.




*hApps register subscriptions with PBV through the Bitcoin interoperability precompiles, and PBV tracks these subscriptions whenever it performs a state update.*

This call registers the subscription with the PBV. When new Bitcoin blocks are processed by the PBV, all active subscriptions are checked to see which ones are triggered by the state transitions the new Bitcoin blocks cause.

When one of the active subscriptions is triggered, the hVM automatically generates a callback transaction. This calls the specified smart contract method with the subscription UUID and the Bitcoin data relevant to the event. These callback transactions are processed in the Hemi block where the PBV state transition occurred, and the gas fees are paid directly by the hApp.




*When PBV performs a state update that triggers an active subscription, PBV executes a callback transaction which notifies the subscribed hApp of the Bitcoin event.*

Because Bitcoin event triggers are deterministically generated by PBV state transitions, these callback transactions are implicit and do not incur DA fees because they are not rolled up to Ethereum.

Smart contracts can subscribe to Bitcoin events involving:

- incoming/outgoing transactions to/from a specific address or scripthash;
- transactions included in chain or reaching a specific confirmation threshold;
- specific UTXOs being spent;
- new Bitcoin blocks being mined;
- existing blocks or transactions being reorged;
- specific fee percentage levels crossing over/under a threshold;
- transfers of a specific satoshi;

- incoming/outgoing/minting inscriptions to/from a specific address or script;
- inscriptions being minted on specific satoshi; and
- incoming/outgoing BRC-20/Rune tokens to/from a specific address or script.

## Bitcoin Tunnel Implementation

The granular Bitcoin data access enabled by the hVM and Hemi Bitcoin Kit provides the foundation for implementing the Hemi Bitcoin Tunnel. It also enables third parties to launch their own Bitcoin Tunnel protocols that optimize security, cost, and speed tradeoffs to serve particular use cases and take advantage of future technical innovations on Bitcoin.

The standard Bitcoin Tunnel on Hemi optimizes for maximum security and capital efficiency, providing high-speed deposits and withdrawals using liquidity providers who swap native and tunneled BTC in an incentive-aligned system monitored by the tunnel protocol.

The Tunnel will be based on a dual-custodianship model:

- **High-Value Vaults**, providing a 1-of-*n* trust assumption for the security of tunneled BTC and other high-value assets based on BitVM [20]
- **Low-Value Vaults**, providing an incentive-aligned trust assumption for the security of lower-value assets where the economics of BitVM are unrealistic

This construction provides all users with 1-of-*n* security guarantees for popular assets regardless of the quantity they own, while providing an alternate system with incentive-aligned security for all other Bitcoin-based assets.

### High-Value Vaults (BitVM)

The high-value vaults will leverage large sets of collateralized validators to establish a BitVM-based custodianship system where large amounts of BTC and other high-value assets can be securely stored. Each validator must stake sufficient funds to reimburse other validators for the Bitcoin fees they would have to pay to prove misbehavior. Note: in this whitepaper we will use the term BitVM to represent BitVM2, the version of BitVM on which we are basing Hemi's tunnel design.

Redeeming native assets from the BitVM custodianship system will be based on a zero-knowledge proof of tunneled asset redemption on Hemi, which will be validated by the BitVM program on Bitcoin in the event of non-unanimous approval of the relevant custodian group. This validation ensures assets are secure as long as one validator is not colluding with the others. Users making significant use of tunneled BTC assets can participate in some or all of the BitVM custodianship groups, making the protocol truly trustless for them.

Deposits to and withdrawals from high-value vaults will be infrequent as a result of BitVM's lengthy (2-4 week) and expensive ($1-10K) withdrawal process in the event of misbehavior. Despite the costs of BitVM, it provides the most capital-efficient method of tunneling large amounts of assets because the 1-of-N trust assumption means far less collateral is required to maintain a secure system.

### Low-Value Vaults (Overcollateralized Custodianship)

Any asset deposit worth less than the cost to properly perform a fault proof on Bitcoin must be deposited to a low-value vault protected by overcollateralized custodians.

Actors can also participate as low-value vault validators by staking sufficient collateral. Validators selected by the protocol to create a new low-fee vault will collectively generate a multisig wallet that requires two-thirds of participants to process a withdrawal.

Each low-security vault can securely manage deposits worth two-thirds of the total assets staked by participating validators. Deposits will be limited to less than two-thirds of the vault value to provide a cushion for asset value appreciation without compromising the vault's economic security. Depositors will specify the value of assets they are depositing and be directed to a vault with sufficient uncommitted collateral. The depositor will pay a fee relative to this claimed value, which will be held in escrow by the tunnel contract. The contract periodically pays a portion of the deposit fee to the vault validators.

The claimed value acts similar to "insurance" — if the native asset is stolen from the vault, the misbehaving validators will be slashed and this value will be given to the owner of the tunneled asset. When an asset is withdrawn, the remaining fees in escrow will be returned to the redeemer.

If the value of assets contained within a vault appreciate to close to two-thirds of the vault's collateral, the tunnel protocol will incentivize holders of the tunneled asset to liquidate a portion of the assets in the vault

using some of the collateral assets to bring the vault's collateral ratio back to an acceptable value.

To abstract this complexity away from end-users, the tunnel protocol will homogenize all fungible assets; for example multiple separate deposits of a specific BRC-20 to different vaults will all mint a common representative token, which can be redeemed for the same underlying asset through any of the vaults.

### Tunnel Liquidity Providers

In order to facilitate faster deposits and withdrawals, tunnel liquidity providers maintain assets natively on Bitcoin and on Hemi and swap them with users in exchange for a fee.

Anyone can register with the tunnel protocol as a liquidity provider if they own a minimum amount of tunneled tokens or the equivalent native token on Bitcoin:

- **Deposit Liquidity**: Owners of tunneled assets will lock them up with the liquidity provider contract on Hemi and register their Bitcoin address. The tunnel protocol will monitor their BTC address for incoming transactions and automatically release an equivalent amount (minus fees) of their tunneled assets to users who send the appropriate native assets to the liquidity provider.
- **Withdrawal Liquidity**: Owners of native assets on Bitcoin will register their Bitcoin address with the liquidity provider contract and stake a small amount of collateral. When a user withdraws, they will send the tunneled asset to the tunnel contract which will assign a liquidity provider to complete the withdrawal. The tunnel protocol will monitor the liquidity provider's BTC address to ensure the withdrawal is processed successfully and then release the tunneled assets (including fees) to the liquidity provider. If the liquidity provider does not fulfill the withdrawal within a specified time period, their collateral will be slashed and the transaction will be re-routed to another liquidity provider.

## Bitcoin Security Inheritance: PoP Mining

### Mining Process

The Bitcoin-Secure Sequencer (BSS) constructs Hemi blocks at a regular, predictable rate. It combines transactions from Hemi's mempool, transactions taken from the Ethereum mainnet, and new Bitcoin headers to create Hemi blocks. Decentralized miners do block construction through a dual-chain rollup block derivation process, allowing tight coupling between Hemi and Ethereum. Every several Hemi blocks,

the network of Pop Miners takes a recent Hemi header and publishes it in a Bitcoin transaction. (See the section on “keystone interval” below.)

The Pop Miner constructs a Bitcoin transaction and propagates it to the Bitcoin network via Bitcoin Finality Governor (BFG) nodes, paying Bitcoin transaction fees to get into one of the next several Bitcoin blocks. Once the Bitcoin network confirms a PoP transaction in a Bitcoin block BFG nodes update their Bitcoin finality statistics and hVM nodes perform PoP payouts.

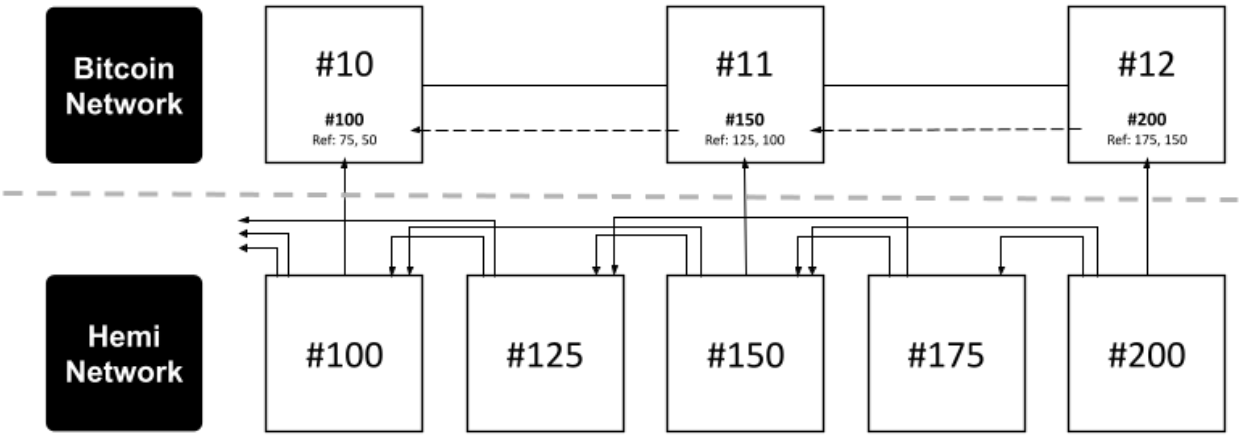
In addition to the standard state data normally published by rollups, Publishers fetch these PoP transactions along with Bitcoin consensus data from the BSS network and publish it to Ethereum via the Hemi Network validation contracts on Ethereum. As a result, the Ethereum-side validation contracts maintain a complete consensus view of Hemi and allow transfers and other smart-contract calls from Hemi to Ethereum to finalize and execute once they achieve Bitcoin finality.

## Protocol Mechanics and Parameter Selection

The following parameters affect how often Hemi headers must be published to the Bitcoin blockchain to maintain full security, when to distribute mining rewards to Pop Miners, and how quickly Hemi reaches Bitcoin finality and, ultimately, Superfinality.

### Keystone Interval (10 minutes)

The Hemi chain is divided into keystone periods, which are discrete chain segments for Bitcoin security inheritance. The keystone interval parameter refers to how many blocks exist in each keystone period. Each first block in a keystone period references the two previous keystone blocks in a braided reference pattern. To maintain the chain’s Bitcoin security inheritance, the system requires only one BTC publication every two keystone intervals.




*Each Hemi keystone references previous keystones to provide full chain visibility from a small number of PoP publications.*

The network does not require Hemi publication to each Bitcoin block. Due to the protocol's design, it can maintain full security while only getting publications into a small minority of Bitcoin blocks. This allows Hemi to continue secure operations even if most of the Bitcoin miners censor fee-competitive PoP transactions from their blocks.

### Bitcoin Finality Delay (9 blocks)

Fluctuations in Bitcoin fees due to variable demand along with occasional long block times can lead to several Bitcoin blocks in a row with no PoP transactions. As a result, a finality delay of nine BTC blocks was selected to ensure the network could maintain robust security regardless of unexpected publication delays from fee spikes or Bitcoin miner censorship. Once the finality delay is reached, Hemi blocks that do not have an actively-mined competing fork published to the Bitcoin blockchain are final and cannot be reversed without 51% attacking the Bitcoin network.

This finality delay prevents edge cases where an attacker with significant consensus control could forcibly violate the "legitimate" chain's finality even if a large minority of Bitcoin miners are actively colluding.

### Rewards

The PoP rewards paid out to Pop Miners must adequately cover the transaction fees of several Bitcoin PoP publications per keystone period. This protects against Bitcoin fee spikes.

The exact per-PoP reward will algorithmically float over time in response to fluctuations in the Bitcoin fee market which the Hemi protocol samples based on PoP publication frequency rather than on-chain fees to

correctly account for external Bitcoin fee markets. The protocol splits this reward among all active Pop Miners based on their relative publication speeds to Bitcoin. This results in an antifragile system resistant to attacks trying to manipulate PoP reward economics.

The protocol only rewards the publication of every other keystone block to the Bitcoin blockchain, so the entire Hemi PoP-allocated security budget can be used to pay for the publications of these consensus-critical blocks.

### Publication Delay Reward Multiplier Curve

For Hemi to incentivize rapid publication of keystone periods in Bitcoin blocks, the protocol applies a scoring penalty to PoP transactions based on the timeliness of inclusion compared to the first publication of the same keystone period in Bitcoin based on the formula $x^{-2}$ (with a special case for x=0).

As a result, PoP transactions for a given keystone period within the first two Bitcoin blocks receive the same reward and have the same consensus value. This design choice prevents Bitcoin miners from being able to benefit from censoring legitimate PoP transactions and inserting their own. If a Bitcoin miner engaged in PoP censorship with this design, they would miss out on profit they would have earned from including PoP transactions correctly. Such an attack would still not threaten Hemi's security because censoring Bitcoin miners are still inserting the consensus-critical transactions themselves. Structuring economic incentives in this fashion ensures decentralized PoP mining participation.

### Early Attack Detection

After the Bitcoin finality delay, the protocol marks with Bitcoin finality the Hemi keystone periods that do not have an active competing fork published to the Bitcoin blockchain. It is mathematically impossible to reverse them without simultaneously 51% attacking Bitcoin and Hemi's native consensus.

The Bitcoin Finality Governor (BFG) scans the Bitcoin blockchain to find any potential forks in Hemi (based on PoP publications) and detects where the potential forks began. It will wait until the attack is resolved to mark any challenged chain segments with Bitcoin finality which occurs when the attacker stops producing new competing keystones and publishing them to the Bitcoin blockchain for nine Bitcoin blocks.

While waiting for full Bitcoin finality, each additional Bitcoin block makes past keystone periods progressively more secure before full finality, requiring an increasing supermajority of Hemi consensus power to successfully reorganize.

The BFG daemon reports the number of Bitcoin confirmations (and, upon the ninth confirmation, full Bitcoin finality) to services like exchanges and payment processors that rely on it for security statistics when confirming transactions.

## Chainbuilder: Extending Bitcoin Security and Interoperability to Other Blockchains

Hemi is designed to serve as a scalable blockchain security root providing Bitcoin Superfinality to other blockchains regardless of their native consensus protocol, architecture, or features.

Blockchains that elect to launch as a Hemi ecosystem chain (hChain) will also inherit fast, secure access to Hemi's robust Bitcoin and Ethereum interoperability.

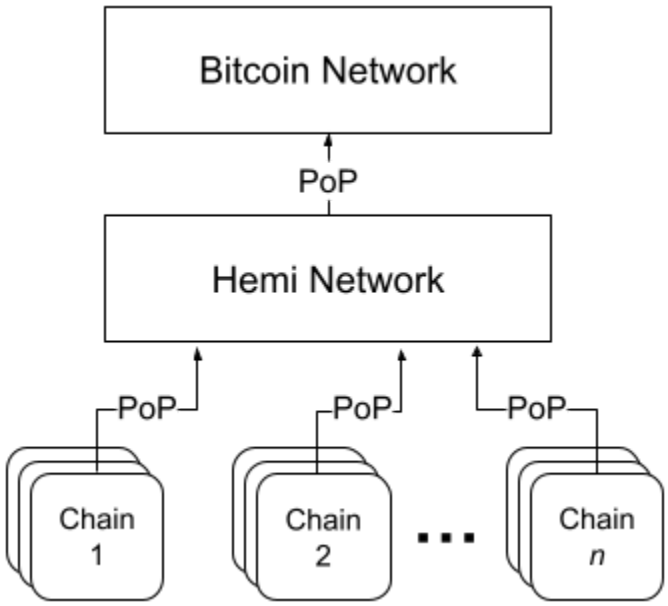
For example, an existing independent blockchain ecosystem could leverage cost-effective Bitcoin security inheritance from Hemi while maintaining full autonomy. Or a new chain could join the Hemi ecosystem as a hChain and provide their users secure access to Bitcoin and Ethereum assets and robust cross-chain connectivity to all other participating chains.

Hemi's Chainbuilder service will make it easy for project teams to launch custom hChains with different execution, consensus, and data availability layers and settlement mechanisms that optimize for specific use-cases that demand particular cost, performance, decentralization, and native security guarantees.

### PoP Security Aggregation

Increasing demand for Bitcoin blockspace makes independent adoption of Bitcoin security by an ever-growing list of blockchains prohibitively expensive and impractical.

Hemi solves this issue by incorporating robust security inheritance and aggregation features directly into its protocol, enabling it to offer Bitcoin-Security-as-a-Service (BSaaS) to the entire blockchain ecosystem. This design enables any blockchain to inherit Bitcoin security through Hemi in a decentralized, permissionless, and cost-effective manner.




*Other blockchains can efficiently inherit Bitcoin security by securing themselves to Hemi using PoP.*

Each chain adds PoP onto its existing consensus protocol of choice (PoW, BFT-style PoS, ETH-style PoS, DPoS, PoST, etc.), and its Pop Miners spend Hemi's native token to publish chain consensus state to the Hemi Network.

The spent native token fees are allocated to Hemi's PoP mining rewards. This process combines the bidding power of all secured chains to guarantee the economic viability of its security aggregation service in the increasingly competitive Bitcoin fee market.

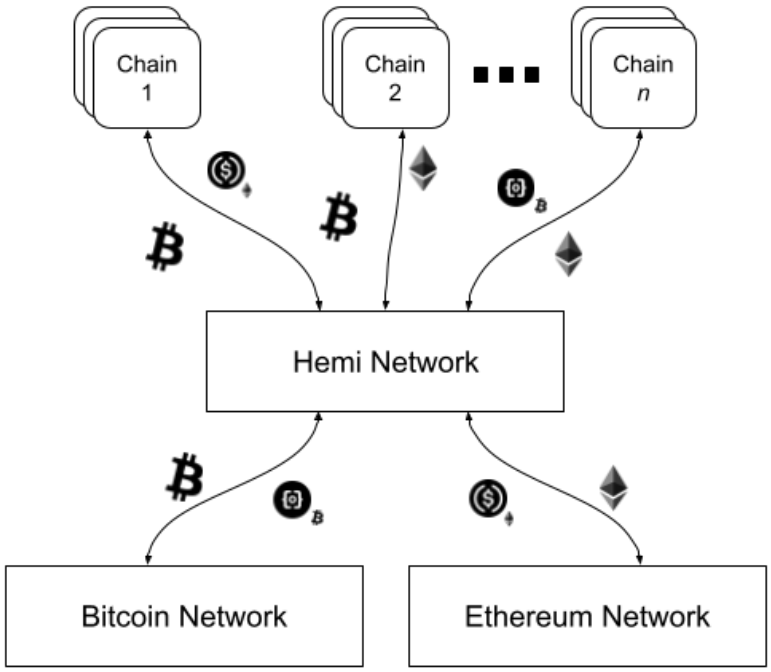
By inheriting Bitcoin security through Hemi and providing Bitcoin finality to all transactions on their network, blockchains protect themselves from 51% attacks regardless of the security of their native consensus protocol.

## Extending Bitcoin and Ethereum Interoperability

In addition to inheriting Bitcoin security, blockchains that elect to launch as hChains enjoy permissionless and secure access to all Bitcoin and Ethereum interoperability features available to smart contracts living natively on Hemi.

All cross-chain contract calls, deposits, and withdrawals between a rollup to Hemi, Ethereum, and other Hemi rollup chains are secured with full Bitcoin finality.

Through the rollup mechanism, rollup networks also have access to tunneled Bitcoin assets and synchronized EVM-level Bitcoin chain state awareness, enabling secure and sophisticated Bitcoin interoperability on their chains.




*Hemi extends Bitcoin and Ethereum interoperability to any chain that deploys on it.*

# Understanding Hemi Decentralized Applications (hApps)

At launch, Hemi will offer native hApps based on well-tested open source projects that utilize Hemi's access to Bitcoin- and Ethereum-based assets, including a lending platform, decentralized exchange (DEX), and DeFi utilities (Merkle claims, payment stream, descending price auctions).

Over time, the suite of hApps will be expanded to include a stablecoin, yield aggregation, and synthetics. Additionally, the team anticipates that basic capabilities for supporting artificial intelligence applications will be available by its mainnet launch.

These default applications are economically integrated with Hemi but do not exert any special privileges over other hApps that developers might want to build.

## Encapsulation

Additionally, one of Hemi's default hApps — Encapsulation — features a multitude of advanced asset-handling capabilities tailored to enhancing platform accessibility and offering a better overall user experience. The goal of this hApp is to provide users and developers the ability to manage and transfer assets more efficiently and securely.

Features will include:

- **Multi-Asset Packaging** – Batch, store, and transfer multiple asset types into a single digital asset.
- **Enhanced Security** – Apply on- and off-chain security — including password protection, time lock, and asset key verification — to any transfer.
- **Smart Routing** – Program and automate multi-step transactions capable of recall or reroute.
- **Gasless Transfer** – Transfer assets without any native chain currency, with or without a web3 wallet.

# Use Cases

By functioning as an interoperability fabric between the world's two largest blockchain ecosystems, Hemi unlocks a number of novel end-user applications and Bitcoin DeFi infrastructure primitives that were previously too error-prone or too expensive in terms of network transaction costs. Some of these are explored below.

## Bitcoin-centric

Applications include:

- **Trustless Bitcoin (Re)Staking Protocols** – Users stake native or tunneled bitcoin to create a collateralized validator marketplace for other protocols and chains; a Bitcoin-based expression of the EigenLayer[23] concept.
- **Bitcoin-authenticated AI Model Marketplace** – Proprietary AI model owners can publish hashes of their model weights to Hemi (thus timestamping to Bitcoin) and authenticate inference outputs for buyers' private inputs through neural-network-optimized ZK-proof systems like zkml[24].
- **Configurable "Smart" BTC Wallets** – These are smart-contract-controlled BTC wallets managed by a configurable validator setup capable of handling native BTC, Ordinals, and BRC-20 assets.
- **Non-Custodial BTC Escrow Services** – Collateralized escrow agents participate in multisig wallets to facilitate the secure movement of Bitcoin-based assets between other parties; useful for lending and creating financial constructs with native delivery.
- **Bitcoin MEV Marketplaces** – Users can pay Bitcoin miners in Hemi or Ethereum assets to process Bitcoin transactions and request specific transaction placement to extract value from meta-protocols, guarantee priority in BRC-20 mints, etc.

## Intersection of Bitcoin and Ethereum

This set of applications includes:

- **BTC / Ordinals / BRC-20 Exchanges –** Users trade Ethereum, Hemi, and Bitcoin assets (native or tunneled) in a single marketplace.
- **Non-Custodial Bitcoin-Enabled Lending Markets** – Users borrow or lend native or tunneled Bitcoin assets against Hemi or Ethereum asset collateral.
- **Bitcoin-Based Financial Instruments** – Developers build financial primitives like options agreements and futures paid for in Ethereum or Hemi assets with settlement in native bitcoin.

Beyond the use cases outlined above, the granularity of Bitcoin data exposed by hVM empowers developers to take advantage of new features in their Hemi apps as they are introduced to Bitcoin. This includes improvements to the core protocol or new innovations on top of Bitcoin like BitVM.

# Summary

Hemi approaches Bitcoin and Ethereum in a novel way – unifying them as components of a single supernetwork rather than treating them as separate siloed ecosystems. This unique architecture combines the individual strengths of crypto's premier economic and technological titans to forge a whole greater than the sum of its parts.

The core of this supernetwork — the Hemi Virtual Machine (hVM) — fuses Ethereum's programmability with robust Bitcoin state awareness delivered by an indexed Bitcoin full node running inside the EVM. This new paradigm of direct Bitcoin introspection — made accessible through the Hemi Bitcoin Kit (hBK) — empowers builders to architect bespoke interoperability infrastructure and develop the next generation of secure and trustless Bitcoin+Ethereum hApps.

Hemi's Tunnel system provides seamless access to Bitcoin and Ethereum assets by facilitating secure settlement to both networks without relying on centralized third parties. Beyond making these assets available to Hemi hApps, the protocol's unique multi-network settlement capability positions Hemi as a conduit *between* Bitcoin and Ethereum by enabling the tunneling of Bitcoin assets *through* Hemi into the broader Ethereum ecosystem.

Combining truly decentralized and permissionless Bitcoin security inheritance powered by Proof-of-Proof (PoP) with data availability on Ethereum enables secure Sequencer, Publisher, and Challenger decentralization while providing accelerated settlement to both chains.

Scaling Hemi's features beyond the capacity of a single blockchain, Chainbuilder enables teams to launch hChains which leverage Hemi's Bitcoin-Security-as-a-Service (BSaaS) capabilities and enjoy access to Bitcoin and Ethereum out-of-the-box. Chainbuilder's modular architecture enables each hChain to customize its execution, consensus, and data availability layers to optimize for diverse use cases that demand specific cost, decentralization, and security guarantees while leveraging Hemi as an efficient security and interoperability aggregation layer.

The evolution of technologies like artificial intelligence into pervasive mainstays of our digital experience necessitates a scalable foundation of trust, provenance, and censorship resistance powered by the combined forces of crypto's preeminent networks.

Integrating Bitcoin and Ethereum into a secure, extensible supernetwork creates the conditions for bringing blockchain technology closer to the status of accessible, pervasive, and critical infrastructure capable of powering tomorrow's Internet and unlocking previously untapped economic, technological, and societal value.

Hemi anticipates a future where the distinction between Bitcoin and Ethereum — as well as "blockchain" and "the Internet" — matters a little less; a future where everyone can enjoy the benefits of these technologies orchestrated together and freed from the complexity that today impedes broader adoption.

# References


(1) S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008

(2) V. Buterin, "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform," 2014

(3) M. Sanchez, et al, "Proof-of-Proof and VeriBlock Blockchain Protocol Consensus Algorithm and Economic Incentivization Specifications," Rev. 2019

(4) Ethereum GitHub, "btcrelay," retrieved 2024-02-29

(5) EF Core Team, "Be Your Own Bank: BeL2 = Bitcoin Elastos Layer 2," Dec. 1. 2023

(6) InternetComputer.Org, "Bitcoin Integration," retrieved 2024-02-29

(7) Ethereum.org, "Optimistic Rollups," retrieved 2024-02-10

(8) Ethereum.org, "Zero-Knowledge Rollups," retrieved 2024-02-10

(9) Ethereum.org, "Validiums," retrieved 2024-06-17

(10) Ethereum.org, "Sidechains," retrieved 2024-02-10

(11) A. Back, et al, "Enabling Blockchain Innovations with Pegged Sidechains," 2014

(12) P. Sztorc, et al, "Hashrate Escrows (Consensus layer)," retrieved 2024-02-10

(13) J. Poon, et al, "The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments," 2014

(14) A. Judmayer, et al, "Merged Mining: Curse or Cure?", 2017

(15) P. Sztorc, et al, "Blind Merged Mining (Consensus layer) Bitcoin Improvement Proposal," retrieved 2024-02-10

(16) Multiple community contributors, "[Counterparty] Protocol Specification," retrieved 2024-02-10

(17) M. Ali, et al, "PoX: Proof of Transfer Mining with Bitcoin," 2020

(18) Optimism, "OP Stack Docs," retrieved 2024-02-10

(19) R. Linus, "BitVM: Compute Anything on Bitcoin," 2023-12-12

(20) E. Frangella, et al, "Aave v3 Technical Paper," 2022

(21) H. Adams, et al, "Uniswap v3 Core," 2021

(22) EigenLayer Whitepaper, Rev. 2024-02-17

(23) D. Kang, "Bridging the Gap: How ZK-SNARKs Bring Transparency to Private ML Models with zkml," 2023-04-12