GSR

November, 2024

Scaling Bitcoin

www.gsr.io

Carlos Guzman, Research Analyst
Toe Bautista, Research Analyst
Brian Rudick, Senior Strategist

Acknowledgements: We'd like to thank Eli Ben-Sasson (StarkWare), Tuğçe Smith (Citrea), and Kyle Ellicott (Stacks) for insightful conversations that informed this report.

While Bitcoin was originally conceived as a peer-to-peer electronic cash system, technical limitations and a fixed supply have seen it morph into a store of value. Renewed efforts to scale Bitcoin and add greater functionality through Layer 2s could see it take on a new role as a settlement layer for decentralized applications, unlocking an array of new possibilities for crypto's largest and most recognizable asset.

Note: GSR is long BTC.

Satoshi Nakamoto, the anonymous developer of Bitcoin, first released the Bitcoin whitepaper in 2008 describing a "peer-to-peer electronic cash system." The ideas in the whitepaper have since spawned an entire industry, which has aimed to remake money, financial services and the internet with key benefits like decentralization, censorship resistance, immutability, permissionlessness, and pseudonymity, among others. Since its introduction, Bitcoin has become one of the world's most valuable assets, assuming the role of a digital store of value thanks to its provable scarcity, decentralization and security.

While Bitcoin is the oldest, best-known and most valuable crypto asset, it's also quite limited. Bitcoin was envisioned as a payments network and its functionality has adhered to that vision. While the blockchain has a basic scripting language that can be used to place conditions on spending BTC, it only supports a small range of use cases like multi-signature wallets and time locks. The network is also slow, handling only up to 5 to 7 transactions per second, which can cause transaction fees to spike to upwards of $40 in times of high demand. This has limited Bitcoin's usefulness even for payments, relegating it to a store of value.

Other ecosystems like Ethereum and Solana have introduced Turing complete programmability via smart contracts that enable decentralized applications, and have innovated to scale transaction throughput. The Bitcoin community, on the other hand, has remained steadfast in its vision to keep the blockchain simple, decentralized, and secure. It has opted not to add Turing complete smart contracts to prevent introducing unintended vulnerabilities or opportunities for MEV that could lead to centralization. Similarly, it's chosen to keep the network throughput manageable for low-powered nodes to favor decentralization.

Bitcoin Layer 2s, however, promise to enhance Bitcoin's functionality and throughput while maintaining the network's security and decentralization, unlocking a suite of potential benefits. This added functionality can allow $1.8 trillion of mostly idle capital to become productive through DeFi, payments, and other decentralized applications. Through L2s, these applications

can benefit from the security and decentralization of Bitcoin, arguably the most secure and decentralized blockchain ever created. Lastly, increased functionality can help solve Bitcoin's declining security budget, an Achilles heel stemming from its capped supply and block reward halvings.

In what follows, we provide an overview of this evolving landscape and some of its latest developments. We start with a brief review on trust assumptions, given its importance to L2 scaling solutions, before exploring efforts around rollups which include work on BitVM, OP_CAT and Data Availability (DA). We then discuss sidechains and finally turn to the long tail of alternative methods beyond sidechains and rollups. It's worth noting that the lines between different types of solutions can be blurry, especially with some early stage projects starting out as one type but aiming to become another. We categorize projects below based on what we think is most appropriate given their current state and end design goals.

A Note on Trust and Layer 2s

The scaling solutions we cover in this report are often referred to as layer 2s, which is sometimes controversial. Strictly defined, a layer 2 network provides added functionality to its layer 1 without introducing new trust assumptions[1]. In reality, almost no chain called a layer 2 today meets this definition. It is in fact widely believed that fully fledged L2 solutions with generalized programmability and trustless bridging are not possible on Bitcoin currently, barring upgrades to the Bitcoin L1. With the above in mind, achieving L2 status is more of an aspiration for the projects we discuss. In practice, they aim to scale Bitcoin's capabilities while minimizing required trust assumptions within the constraints of what is possible today.

The two most important areas where trust assumptions are required in Bitcoin relate to 1) the trust assumptions required to ensure fund safety, and 2) those required to ensure double-spend resistance. Simplifying a bit, ensuring fund safety in Bitcoin does not require trust in external parties, since anyone can run a node and verify the correctness of transactions included in the blockchain directly. For double spend resistance, users have to trust that more than half of the network's hashpower is in honest hands, since the majority will always build the longest chain and can thus prevent transactions from being reverted.

Bitcoin L2s aim to minimize the trust assumptions they add on top of these two, striving to inherit the L1's security to the highest degree possible. There's an added complication in the case of L2s in that most solutions require bridging to move BTC from the L1 to the L2 and back. Bridging is accomplished by locking BTC on the L1 and minting a token representing this locked BTC on L2. Once users want to bridge back to L1, they burn the L2 token and redeem it for L1 BTC. Ensuring the security of the funds locked in L1 is thus of paramount importance for L2s. To date, bridging has mostly been handled through multisigs, where a set of trusted authorities

[1] Note: We'll often speak of trust assumptions by referring to the number of parties *n* that need to be trusted out of a total set of *m* parties. We provide more context on how to interpret this in the appendix.

custody the funds on L1 and process withdrawals. The aim for most L2s is to improve upon this model and achieve greater trust minimization.

Ideally, then, an L2 should:

1. Inherit Bitcoin's double-spend resistance
2. Enable the security of funds in the L2 to be verified by Bitcoin L1 nodes
3. Have trustless bridging, where users can simply withdraw funds from the bridge by submitting a transaction on L1

Practically no current L2 solutions achieve full trustlessness across all three areas, but instead offer different tradeoffs. We survey those tradeoffs in what follows.

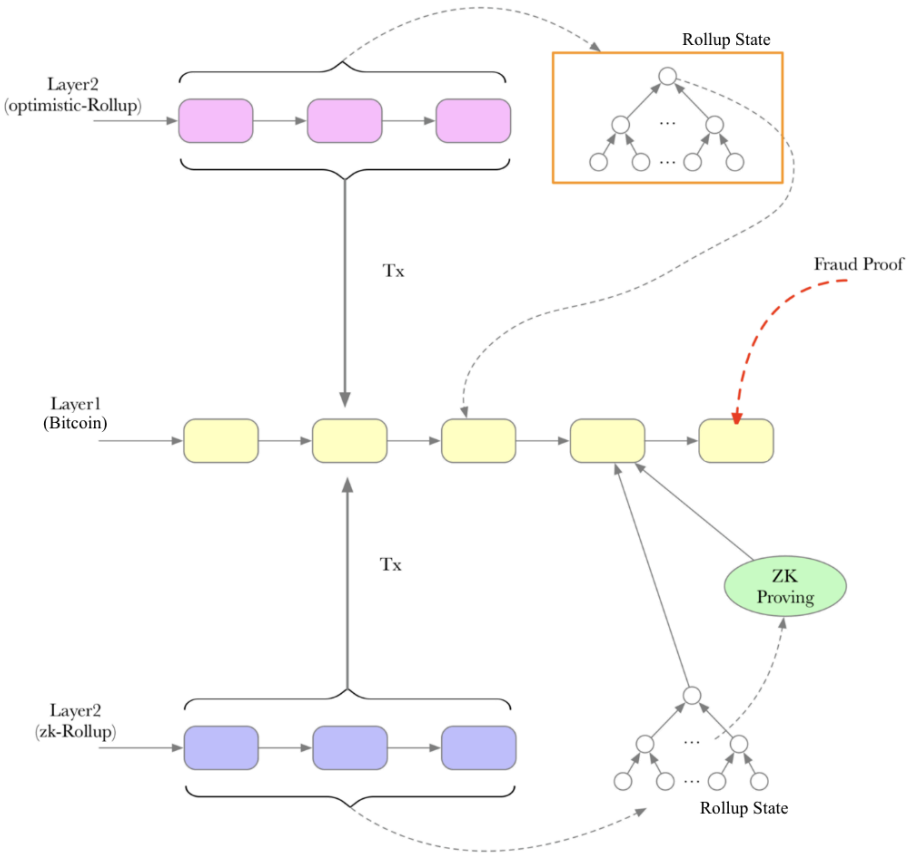
Rollups

A rollup operates as a separate chain from the L1, but inherits its security by providing the L1 with proofs of correctness. Rollups collect L2 transactions in rollup blocks and post them to the L1. They also provide a proof attesting to the correctness of the transactions. By having the L1 verify proofs rather than re-execute all transactions, rollups can scale transaction throughput and enable arbitrary programmability.

Rollups come in two flavors, zero-knowledge (zk) and optimistic. ZK rollups provide direct proofs of validity using the mathematical properties of zkSNARKS, which can be used by one party (a prover) to convince another (a verifier) of the correct execution of a computation with a proof that is quicker to verify than re-executing the computation itself. In the case of optimistic rollups, a challenge period exists during which another party can dispute the validity of rollup transactions with a fraud proof, and transactions are assumed to be valid once the challenge period has passed and no challenges have been made.

Both types of proofs enable L1 nodes to validate the rollup's state transitions, thus verifying the safety of funds in the L2. Once a proof is verified, or a challenge period has passed, the L1 will update its view of the rollup's latest state. If users want to withdraw funds back to L1, they can use the latest rollup state to provide proof that they own funds on the L2, allowing them to withdraw them.

In theory, rollups achieve trustlessness across all three prongs mentioned previously, as they 1) inherit the L1's double-spend resistance, since the canonical rollup chain follows the data posted to L1, 2) they inherit the L1's safety as L1 nodes verify the correctness of state transitions via proofs and 3) users are able to unilaterally withdraw their funds.



Source: msfew.eth, GSR.

Rollups have traditionally been associated with the Ethereum ecosystem, and until recently were thought to be impossible to implement on Bitcoin. Some of the recent excitement around Bitcoin L2s comes from the realization that Bitcoin rollups may now be possible.

Bitcoin's 2021 Taproot upgrade played a large role in making this a reality as it effectively removed limits on the amount of arbitrary data that could be inscribed into Bitcoin blocks, up to the full block data limit of 4MB. This change made rollups on Bitcoin more feasible, as rollup transaction data could now be posted on the L1.

This by itself made one flavor of rollups, so-called 'sovereign rollups', possible on Bitcoin. Sovereign rollups only use the L1 for consensus and double-spend resistance, but don't rely on the L1 to determine their canonical state, and therefore usually do not support trustless bridging. Projects like Bison Labs, Sovereign Labs and Celestia's Rollkit are working on frameworks to build sovereign rollups on Bitcoin.

Building fully-fledged Bitcoin rollups with trustless bridges, however, requires more than posting data to L1. Two further features are required:

1. Verification of L2 state transitions on L1
2. Persistent programmable locking and withdrawing of funds on L1

Neither of these has traditionally been thought possible to do on Bitcoin as it is today. The first requires Bitcoin to verify either a zk proof or a fraud proof – however, Bitcoin's limited programming language does not natively support the range of computations needed to verify either type of proof. The second also requires greater programmability, allowing the creation of smart contracts that can control funds indefinitely. Recent research breakthroughs and possible upcoming upgrades to Bitcoin promise to unlock both of these features, making trustless rollup bridges possible. The two that have received the most attention are BitVM and the re-enablement of the OP_CAT opcode.

BitVM

BitVM is a system for enabling arbitrary computation (so-called Turing-complete computation) on Bitcoin. This expands the range of code that can be executed on Bitcoin beyond the bounds of its limited scripting language. Contrary to what the name might suggest, BitVM is not directly analogous to the Ethereum Virtual Machine (EVM), which executes programs onchain. Instead, BitVM allows for arbitrary computation to be performed off-chain while providing a mechanism to challenge the validity of that computation on Bitcoin if needed. It thus enables optimistic verification similar to fraud proof mechanisms used by optimistic rollups on Ethereum. While BitVM is not itself a rollup, it provides a way for rollups to have their state transitions verified by Bitcoin.

A key feature of BitVM is that it does not require an upgrade to Bitcoin. It generated much buzz when it was introduced by Robin Linus in late 2023, as it was previously believed that Turing-complete computation was not possible on the L1. Linus' initial whitepaper showed how you could in principle implement a challenge-verification game for Turing-complete computation using existing Bitcoin functionality like hashlocks, timelocks and large Taproot trees.

However, the initial version of BitVM wasn't practical. It implemented computation at an extremely low level, using two Bitcoin opcodes to implement basic logic gates. Furthermore, BitVM's challenge mechanism could in the worst case take up to 6 months to resolve, creating an impractically long waiting period for trustless bridging. The protocol also allowed only two parties, a prover and a verifier, to participate in a dispute game, resulting in a sub-optimal trust setup (1 of 2 trust assumption).

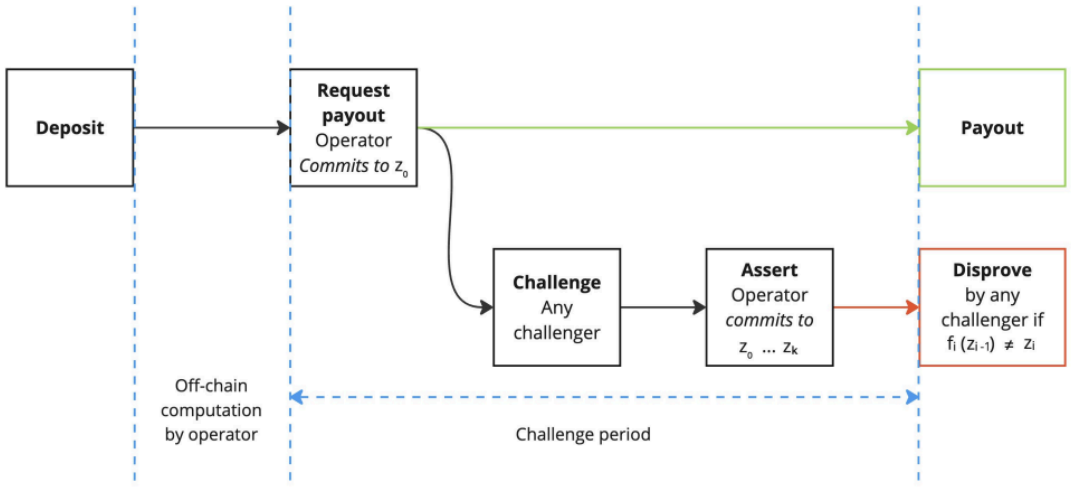
Since the introduction of BitVM, several contributors have been working on more practical versions of the protocol. The most recent design, BitVM2, proposed by Linus and others including the team behind BOB, greatly improves upon BitVM and addresses many of its drawbacks. BitVM2 improves on BitVM's two-party limitation by enabling permissionless

challenging such that anyone can challenge an incorrect computation. Disputes in BitVM2 also only require one round of just three Bitcoin transactions, thus reducing the time to resolution down to 1-2 weeks instead of the worst case of 6 months for BitVM.

BitVM2 accomplishes this by having the prover commit to correctly executing a zk proof verifier off-chain. Challengers can then dispute the correct execution of this verifier. ZK proof verifiers can be used to verify arbitrary computation, thanks to the existence of Turing-complete zkVMs. Thus, BitVM2 still enables Bitcoin to verify Turing-complete computation while simplifying the scope of the protocol.

While we don't yet have production-ready implementations of BitVM2, the protocol's design is viewed as more practical and the researchers behind it are currently working on optimizations to reduce onchain costs and ensure the protocol's security. Other teams have also proposed and are working on protocols that similarly improve upon BitVM1 by reducing the protocol's scope to executing a zk proof verifier. Such protocols include Alpen Labs' SNARKnado and BitcoinOS' BitSNARK. The progress being made across these efforts makes us optimistic that a practical BitVM implementation could be ready for mainnet shortly.

BitVM2 Simplifies Challenge Process to Just Three Onchain Transactions



Source: BitVM2 whitepaper, GSR.

While a fully-fledged BitVM implementation will enable Bitcoin to verify rollup state transitions, moving the space closer to fully trustless L2s, BitVM doesn't enable the persistent smart contract functionality needed to create trustless bridges. What are known as 'recursive covenants' are needed for this. Instead, BitVM bridges need an operator to fulfill withdrawals from the bridge. This operator fronts withdrawals to users and then requests reimbursement

using funds locked in the bridge. If the operator is dishonest and requests reimbursement for withdrawals that have not been honored, a challenger can use the BitVM challenge mechanism to prove the operator has misbehaved and prevent them from fraudulently withdrawing funds. BitVM bridges thus enable a 1 of n trust assumption, since either the operator or a challenger must be honest.

These BitVM bridge designs have been criticized due to potential liquidity issues in cases of mass withdrawal, since an operator may not be able to source enough funds to honor all withdrawals within the cutoff period, potentially leading to loss of funds. More recent designs claim to address this drawback by having multiple operators and including priority fee mechanisms, but this remains a potential challenge that production systems will need to address.

Despite its drawbacks, BitVM represents the most promising way of implementing rollups on Bitcoin without having to upgrade the L1. Several projects are working on rollup designs leveraging BitVM, for more detail on these projects see the appendix.

OP_CAT

While BitVM provides a way to support rollups on Bitcoin without upgrading the L1, upgrades to the protocol could make supporting rollups significantly easier and more efficient. Upgrading the Bitcoin protocol is notoriously difficult, however, as it requires building sufficient consensus in a community that values security and decentralization, and thus views changes with a high degree of skepticism. The re-enablement of the OP_CAT opcode is viewed as a promising upgrade that could support rollups on Bitcoin by enabling both zk proof verification and recursive covenants. Proponents argue that enabling OP_CAT would be safe, as it was previously part of Bitcoin and has been successfully implemented in Bitcoin forks like Bitcoin Cash and Liquid without any apparent issues.

OP_CAT is an instruction in Bitcoin's scripting language. The CAT in OP_CAT is short for 'concatenate' – all it does is join two elements into one. The opcode was included in Bitcoin's initial version, however, it was disabled in 2010 by Satoshi due to concerns that it could be used in a denial of service attack on the network. Proposals to reenable OP_CAT include safeguards that address those concerns.

Although OP_CAT is quite simple, it can be cleverly used to unlock additional functionality, such as verification of zk proofs, specifically zkSTARKs. Verifying STARKs requires the creation and verification of Merkle trees, which at its core involves concatenating and hashing values. Since Bitcoin already supports operations for hashing, enabling OP_CAT would add the final ingredient needed to verify STARK proofs directly on Bitcoin, without the need for an off-chain computation gadget like BitVM.

The second key item that OP_CAT unlocks is recursive covenants. Recursive covenants enable persistent spending conditions to be placed on BTC, emulating smart contracts. When BTC is spent today, an unspent transaction output (UTXO) is created which enables the recipient of the funds to spend them in the future as long as they can meet the spending conditions (e.g., signing with the right private key). The conditions needed to spend a UTXO only apply to the spending of that specific UTXO, and don't continue to apply to those funds in the future. Recursive covenants enable UTXOs that can place spending conditions on subsequent UTXOs after they are spent. Using clever tricks that exploit the features of Bitcoin's Shnorr signatures, OP_CAT can be used to enable recursive covenants.

Recursive covenants are key for rollups because they allow for the creation of programmable vaults on Bitcoin L1, which can be used to create trustless bridges. Instead of needing an operator to manage withdrawals from an L2 bridge as is the case with BitVM, a vault would be able to handle this functionality automatically without the need for an intermediary, thus improving on trust assumptions to achieve 0 of n trust.

OP_CAT thus holds great promise to enable rollups on Bitcoin. While other proposals exist to introduce opcodes that verify zk proofs or enable covenants, OP_CAT stands out in that it enables both at once with a simple primitive. Although OP_CAT is an inefficient way of accomplishing these functionalities compared to purpose-built opcodes, it has the benefit of being possible to activate with a simple soft fork. If enabled it is likely that most Bitcoin rollup projects would adopt it to improve their bridge designs.

One team in particular, StarkWare, is worth mentioning with regard to OP_CAT. They've created a $1 million fund to support research on the feasibility and safety of reenabling OP_CAT on Bitcoin. They've further committed to extending their Ethereum rollup, StarkNet, to settle on both Ethereum and Bitcoin within six months of OP_CAT being activated. This would be a full circle moment for StarkWare whose founder Eli Ben-Sasson, the creator of STARKs, proposed applying zk proofs in Bitcoin back in 2013.

Data Availability

A potential drawback that rollup designs on Bitcoin have is related to data availability (DA). As discussed, rollups need to post their transaction and state data, along with their proofs, to the L1. This allows users to reconstruct the rollup state and gives them the data they need to withdraw funds on L1. Unfortunately, the storage space available on Bitcoin to post this data is highly limited. Bitcoin blocks can store at most 4 MB of data, and a new block is generated every ~10 mins. Thus, Bitcoin has a data availability throughput of ~6.67 KB/s. Even ignoring proof size and assuming data per transaction is highly compressed down to 12 bytes, this would enable storage of 500-600 transactions per second for all rollups on Bitcoin, assuming the entirety of Bitcoin blocks are used to store rollup transaction data. While much higher than

Bitcoin's 7 tps max throughput today, this level of throughput would still be insufficient to serve internet-scale use cases, and could thus still result in high fees to transact on rollups.

To avoid these issues, rollups could choose to post their data on an alternative DA solution like Celestia or use a data availability committee (DAC). However, this introduces additional trust assumptions, as users now have to rely on new external parties to ensure the availability of the data they need to verify the rollup. Some projects like Nubit are building Bitcoin-aligned data availability layers, which inherit some of Bitcoin's security properties like its double-spend resistance. While they still introduce some trust assumptions, they can offer a comparatively trust-minimized solution for Bitcoin rollups.

The teams behind rollups like Starknet, Strata, Citrea, and others are also actively researching the implementation of a volition mechanism, which enables apps and/or users to choose where they want particular transaction data to be stored. This can provide a middle-ground, allowing users to choose to pay higher fees if they want their transactions to inherit Bitcoin's full security, or opt to pay less if they're fine with the additional trust assumptions of a different DA layer.

Sidechains

Outside of rollups, sidechains in particular continue to be a popular scaling model. Bitcoin sidechains are not new, as existing projects like Rootstock and Liquid have had live sidechains since 2018. However, projects continue to innovate in this space to improve security and trust assumptions.

Sidechains, as the name implies, are separate blockchains that run parallel to the main chain. Unlike rollups, sidechains don't post their transaction data on the main chain and do not rely on the main chain to verify their state transitions. Instead, sidechains have their own set of nodes and validators, and have their own consensus mechanisms. This means that Bitcoin can't directly validate the sidechain, so security depends on the L2 validators. However, it can enable sidechains to be more cost-effective, since they don't have to post data to the base layer. Sidechains also have the advantage of not having to rely on new and complex mechanisms like BitVM, whose failure modes and security are still the subject of ongoing research. They can instead rely on methods for trust minimization that have been around for longer and whose security properties are better understood.

Since sidechains do not have their state transitions verified by L1, they don't meet prong 2 of our L2 criteria. Most sidechain projects, however, implement mechanisms to minimize trust across prong 1 (inheriting Bitcoin's double-spend resistance) and prong 3 (trust minimized bridging). We discuss these two below.

Double-Spend Resistance and Bitcoin-Aligned Economic Security

Various sidechain projects take different approaches to inheriting Bitcoin's double-spend resistance or otherwise leveraging Bitcoin for their economic security. Economic security here refers to the monetary amount that an adversary would have to spend to execute a double-spend or otherwise compromise a blockchain.

As discussed, rollups inherit Bitcoin's double-spend resistance, and therefore its economic security, by posting their data on Bitcoin. The canonical rollup chain is determined by looking at the data on Bitcoin itself, so it has the same security properties. Many sidechains aim to inherit Bitcoin's economic security in a similar way, but rather than posting full transaction data or state differences, they post commitments to particular transaction histories, usually in the form of state root hashes. Nodes can then compare the data they're getting from other L2 nodes with the commitments posted on L1 to verify that they are following the canonical chain. Botanix, an upcoming sidechain, will use this exact mechanism and expects to post a new state root on every Bitcoin block. Similarly, miners on Stacks commit to the blocks proposed by the previous miner on every Bitcoin block, and Stacks validators (called Stackers) consider blocks irreversible once a state commitment has been confirmed by 150 Bitcoin blocks. These mechanisms have the benefit that they inherit Bitcoin's double-spend resistance for blocks that have been committed, but they have the downside that blocks that haven't yet been committed depend on the sidechain's economic security and can be reorganized.

Another approach some sidechains use to inherit at least a part of Bitcoin's double-spend resistance is 'merged mining', which is used by chains like Rootstock. Merged mining enables Bitcoin miners to opt into mining blocks for a sidechain at the same time as they mine Bitcoin blocks, in exchange for a portion of the sidechain's transaction fees. This enables the hashpower which is used to secure Bitcoin to also be used to secure the sidechain. A drawback of merged mining is that miners can choose whether or not they want to participate, which in practice means that only a fraction of the total hashpower is used to secure the sidechain (42% in Rootstock's case as of the most recent data).

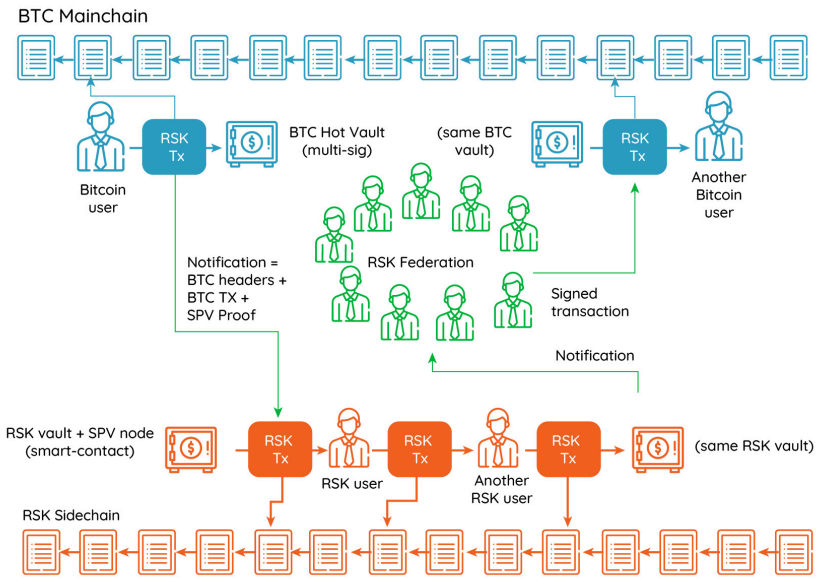
Sidechains that do not use merged mining have to provide economic security for their chain some other way, even if they make state commitments to L1 since those commitments happen only after consensus has been reached on the L2. Many chains accomplish this with their own set of miners or validators staking the chain's native token. An interesting set of projects, however, aims to accomplish this while staying as Bitcoin aligned as possible. This includes Botanix, Mezo and Stacks. Botanix and Mezo are both proof-of-stake chains, however, they both allow validators to stake BTC to secure their chains, thus taking advantage of BTC's value and stability. Stacks uses a mechanism dubbed proof-of-transfer (PoX) whereby miners spend BTC and are selected to propose Stacks blocks with a probability that corresponds to the amount of BTC spent. Both Mezo's and Stacks' designs are hybrids, Mezo allows dual staking of the Mezo token along with BTC, while Stacks only uses BTC proof-of-transfer for block production, with validation handled by 'Stackers' who stake STX to participate as validators.

These mechanisms thus allow these sidechains to draw some of their economic security from Bitcoin and the value of BTC they can attract to stake on their platforms, but they don't quite reach the level of fully inheriting Bitcoin's economic security and double-spend resistance.

Multi-Sig Bridges, Threshold Signatures and Cryptoeconomic Guarantees

Sidechains also seek to minimize trust in bridging through a variety of different mechanisms. Existing Bitcoin sidechains have for the most part relied on a federation of trusted parties to secure the bridge between Bitcoin and the sidechain using multisigs. These usually consist of federations of between 10-15 trusted signers, where an honest majority or supermajority (e.g., 11 of 15) is needed to sign off on withdrawals.

Sidechains Traditionally Rely on Trusted Federations for Bridging



Source: Rootstock, GSR.

Recent sidechain projects have been exploring other models to further minimize bridge trust assumptions. A popular model has been the use of threshold signatures. Threshold signatures are conceptually very similar to multisigs, both enable authorization of actions based on collecting sufficient signatures out of a set of possible signers. Multisigs, however, use individual public keys for each signer in a fixed set of signers. The chain must keep track of these public keys and verify their individual signatures, which means that the onchain footprint of multisigs grows with the number of signers. Threshold signatures, on the other hand, only require a single public key and use Multi-Party Computation (MPC) off-chain to split shares of a single private key across many participants. In order to sign transactions, these participants once again use

MPC off-chain to privately sign using their shares and once a specific threshold number of participants is met, the private key can be recovered and used to sign transactions. Threshold signatures can thus scale to a greater number of participants, allow for more flexible signer sets, and have a lower onchain footprint.

Botanix, Mezo, and Stacks all use threshold signatures as a way of achieving greater trust minimization than multisig federations. All three aim to enable permissionless participation in their signer sets using the cryptoeconomic guarantees of staking to ensure honest participation, rather than relying on the reputation of a small number of trusted parties.

In Stacks' upcoming sBTC bridging mechanism, Stackers will be in charge of operating the bridge, which will require 70% of the stake to sign off on any withdrawals. If Stackers do not process withdrawals, they lose access to their stake as well as the BTC rewards they would otherwise earn for their service. In order to preserve the incentive compatibility of the mechanism, the amount of BTC that can be bridged through the sBTC mechanism is capped at a fraction of the total stake securing the bridge such that attacks on it are not profitable.

Botanix and Mezo also require threshold signers to stake to provide cryptoeconomic security for their bridging mechanisms, but they add a further layer of probabilistic guarantees. Instead of having a single deposit wallet for their bridges, Botanix and Mezo frequently create new deposit wallets (every Bitcoin block for Botanix and every two weeks for Mezo's tBTC). Each of these wallets is secured by a new set of 100 signers drawn from a set of staked participants, with the likelihood of being selected varying according to their stake. This makes it so that even if a malicious actor acquires a large minority portion of the stake, their probability of controlling any deposit wallet is low. Furthermore, even if a malicious actor does come to control a deposit wallet, they will only be able to compromise the limited funds in that wallet while exposing themselves to slashing risk. This mechanism thus provides cryptoeconomic security as well as 'forward security' since the compromise of a deposit wallet does not lead to the compromise of past wallets, given these were created with different signer sets.

Thus, while sidechains do not have the same theoretical security guarantees as rollups that can fully inherit the L1s security, they can still use a variety of mechanisms and cryptoeconomic guarantees to ensure some level of trust-minimization.

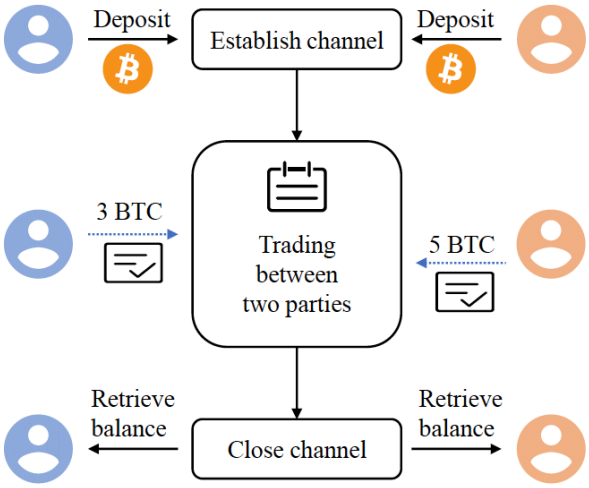
Other Approaches

While rollups and sidechains cover the bulk of L2 projects building on Bitcoin, there are some other approaches that do not fall into these two buckets.

A prominent category is payment channels, as it includes one of the most popular Bitcoin scaling solutions, the Lightning Network. The Lightning Network is arguably the only existing

Bitcoin scaling solution that can accurately be described as a layer 2, as it does not require trust in external parties. It doesn't operate as a separate blockchain, instead users open bi-directional off-chain channels which can be opened by locking funds in 2 of 2 multisigs. Users can then sign off-chain transactions which transfer the funds in the channel, but are not broadcasted to the L1. At any point, users can close the channel and leave with their latest balance by submitting the latest channel state in L1. If either party tries to cheat by submitting an older state, the other party can use a 'revocation secret' to recoup their funds. Despite its benefits, Lightning does have limitations, as it is primarily suited for payments, requires users to run a node and remain online for full security, and can face liquidity and channel routing challenges for large payments. However, efforts like Lightspark aim to build on Lightning, abstracting its complexity and enhancing its utility.

Lightning Allows Users to Transact Trustlessly via Off-Chain Channels



Source: Huawei Huang, GSR.

Projects like Ark attempt to improve upon the Lightning Network by generalizing its basic concepts. Instead of funds being taken to transact off-chain by locking them in a UTXO controlled by two parties, Ark allows this to happen with an arbitrary number of users. The participants can then trade their shares in the existing UTXO for shares in a future UTXO that will be created at the end of the round. While a central party coordinates these transfers, they don't custody funds, as users can unilaterally exit. Like Lightning, Ark can only support payments, but it can be a promising solution for use cases like BTC microtransactions.

Yet another class of approaches is called 'client side validation', such as RGB. Like Lightning, it doesn't involve a separate chain. Instead, users transact peer-to-peer and run specialized client software to validate the history of their transactions and those of their counterparties. RGB enables smart contracts to be performed peer-to-peer on Bitcoin. A drawback of RGB is that it

only inherits Bitcoin's full security and works trustlessly when operating with new assets issued on top of the protocol. In order to use BTC in the protocol it has to be wrapped (bridged), and current designs use a federation to control that mechanism. RGB's smart contracts also only involve the specific parties transacting, so it's unable to support 'ownerless' smart contracts like automated market makers and lending protocols as they exist on Ethereum.

There are also projects that straddle the boundary between L1 and L2 and don't fit into any specific category, such as Arch Network. Arch works by having a network of nodes separate from Bitcoin, which can nonetheless sign Bitcoin transactions using a threshold signature scheme. Users on Bitcoin can make requests to the network to execute arbitrary smart contract code off-chain, which can operate on Bitcoin L1 UTXOs that users provide. If more than 50% of the Arch network nodes sign off on the changes, these are settled back to Bitcoin. Thus, Arch enables smart contracts to operate on BTC without requiring bridging, which works assuming that the majority of the Arch network is honest.

Lastly, Babylon helps unlock Bitcoin's productivity by allowing Bitcoin to be staked to secure other blockchain networks. This allows Bitcoin holders, who take on slashing risk, to earn yield by staking it to validators of other networks, which then pass on validation rewards to stakers after taking a commission. Babylon doesn't enable BTC smart contracts or payments, but it enhances BTC's utility by enabling proof-of-stake chains to rent Bitcoin economic security and provides a way for holders to earn rewards.

–

In sum, the Bitcoin L2 landscape is quickly evolving. Spurred by innovation in other blockchain ecosystems and research breakthroughs like BitVM, parts of the Bitcoin community have come to see the potential that exists in taking crypto's biggest asset and making it productive in decentralized applications, while simultaneously allowing the L2 solutions hosting these applications to inherit Bitcoin's unparalleled decentralization and security.

Bitcoin's limitations have made this vision difficult to realize, but the influx of talented developers and renewed energy make us optimistic that these challenges can be overcome. Developments like BitVM and the potential reenablement of OP_CAT promise to make rollups with trust-minimized bridging possible on Bitcoin, unlocking the use of Bitcoin in fully programmable applications with minimal additional trust assumptions. The story, however, doesn't end with rollups. Sidechain projects have innovated to inherit Bitcoin's double-spend resistance and improve bridge trust assumptions through cryptographic and cryptoeconomic mechanisms. Furthermore, a long tail of projects is working on other approaches including payment channels, client side validation, and even hybrid L1/L2 designs. This diversity of approaches allows the space to explore different sets of tradeoffs, increasing the likelihood that solutions providing the optimal set of trust assumptions for different types of users will be found.

The amount of innovation happening on Bitcoin and the renewed builder interest in the ecosystem is encouraging to see. We believe a massive opportunity exists in unlocking the utility and functionality of Bitcoin and look forward to seeing how the different approaches evolve in pursuit of this goal.

Appendix

Bitcoin Layer 2 Projects

The following is a non-exhaustive list of prominent projects across the L2 categories covered.

Rollups and Rollup Infrastructure:

- Bitlayer**:** Bitlayer is an EVM-compatible Bitcoin project that is live today as a sidechain and has attracted over $450M in TVL. Bitlayer today operates as a proof-of-stake (PoS) chain with a multisig bridge, but intends to upgrade to become a rollup with a trust-minimized bridge once BitVM technology is mature and production-ready.
- BitcoinOS**:** BitcoinOS is building a framework to create a network of EVM-compatible rollups on Bitcoin. It is being built by an alliance of partners, including Sovryn, Merlin, Bsquared Network, Nubit, and Emurgo. The project has released research and designs building upon BitVM, including the BitSNARK protocol and Grail bridge.
- Bison Labs: Bison Labs is building a framework to develop zk-based sovereign rollups on Bitcoin. They are building a trust-minimized bridging solution which will use a network of MPC nodes to verify zk proofs posted to L1 and process withdrawals using a multisig. They envision upgrading this mechanism to use Discreet Log Contracts, which will require both the user and the MPC network to jointly sign for withdrawals to occur.
- BOB (Build on Bitcoin): BOB is building a hybrid L2 that will settle on both Ethereum and Bitcoin (i.e., it will have trust-minimized bridges to both). BOB is built as an EVM-compatible rollup using the OP Stack and will start as an Ethereum rollup before adding a trust-minimized bridge to Bitcoin once BitVM tech is mature. The BOB team is actively contributing to the development of BitVM2.
- Citrea: Citrea is an EVM-compatible Bitcoin rollup. The team behind it has proposed a bespoke BitVM bridge design dubbed Clementine and are working to implement rollup features like pre-confirmations and forced transactions which can offer improved UX and stronger censorship resistance guarantees.
- Nubit: Nubit is building infrastructure to enable Bitcoin-based rollups. They've developed several offerings, but their flagship product is a Bitcoin-aligned Data Availability layer, which enables rollups to post their data at lower cost while enabling some of Bitcoin's economic security through BTC staking and block commitments on L1.

- Rollkit: Rollkit is a sovereign rollup development framework created by the Celestia team. The framework is compatible with Bitcoin as a DA layer, enabling the creation of sovereign rollups on Bitcoin.
- Sovereign Labs: Sovereign Labs is building a framework and other infrastructure to enable sovereign zk rollups. Their framework aims to support rollups building on top of any chain for settlement and DA, which includes Bitcoin.
- Starknet: Starknet is an Ethereum rollup built by Starkware, one of the first developers of zk rollups. It implements the CairoVM, a custom-built VM optimized for STARK proving, and benefits from key advancements in zk proving developed by Starkware including Circle STARKs and the upcoming Stwo prover. Starknet plans to settle on both Ethereum and Bitcoin if OP_CAT is enabled via a Bitcoin soft fork.
- Strata: Strata is an EVM-compatible rollup being built by the team at Alpen Labs. The team has contributed significant research to the space and includes some of the earliest proponents of building zk rollups on Bitcoin. Alpen has proposed improvements to BitVM like SNARKnado and contributes to the development of BitVM2 as part of the BitVM alliance along with Citrea.

Sidechains:

- Botanix: Botanix is building an EVM-compatible Bitcoin sidechain. The chain will use BTC as both its staking and gas token. The project is best known for its 'spiderchain' concept, which describes its bridging mechanism which uses new threshold signature-based deposit wallets created on every Bitcoin block.
- Liquid: Liquid is an early Bitcoin sidechain project developed by the team at Blockstream. The sidechain implements a modified version of Bitcoin which includes several new opcodes, enabling it to support covenants. Liquid extends Bitcoin's functionality to support new asset issuance and confidential transfers. It uses a federated multisig for bridging.
- Mezo: Mezo is a Bitcoin scaling layer developed by Thesis, the team behind popular products like tBTC, Fold and Taho. Mezo is an EVM-compatible chain built on the Cosmos SDK. Mezo allows validators to stake BTC or Mezo's native token MEZO. To further align with Bitcoin, BTC is also used as the gas token.
- Rootstock: Rootstock (RSK for short) was one of the earliest Bitcoin sidechain projects with a mainnet launch in 2018. It provides EVM-compatible smart contract capabilities on top of Bitcoin. The chain uses merged-mining to inherit some of Bitcoin's economic security and has a federated multisig bridge that uses trusted hardware security modules for improved trust guarantees.
- Stacks: Stacks was one of the earliest projects to focus on scaling Bitcoin and extending its functionality. The chain uses a unique proof-of-transfer consensus mechanism which leverages BTC for economic security and they also developed their own decidable smart

contract programming language called clarity. The project has been receiving a makeover recently with its Nakamoto Upgrade, and its upcoming sBTC bridge.

Other Approaches:

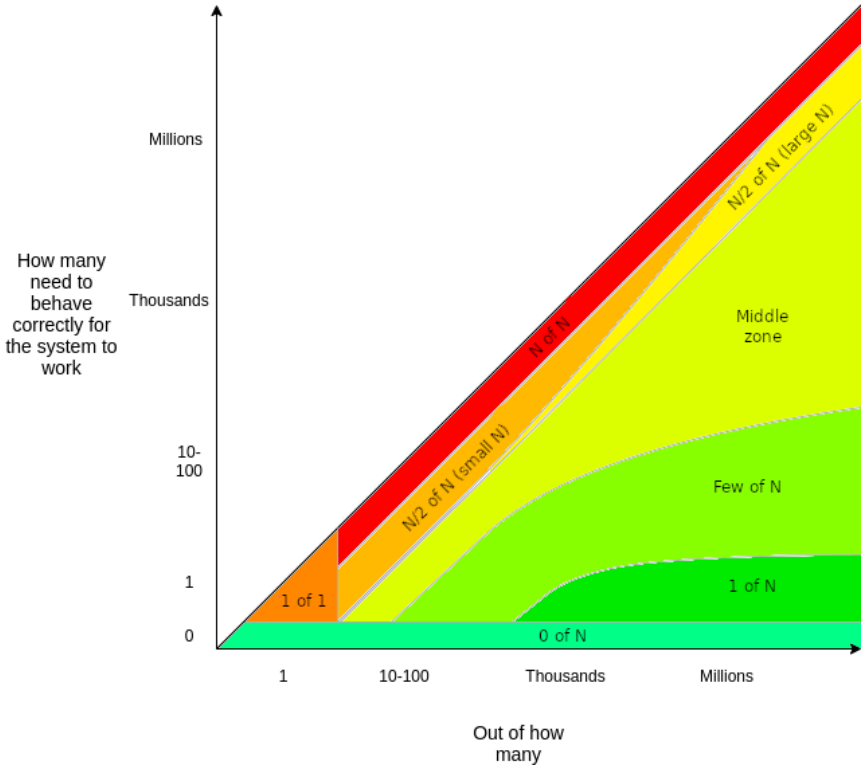
- Arch Network: Arch Network is building a solution to enable the execution of smart contracts on the Bitcoin L1 without requiring funds to be bridged over to an L2. They leverage a network of nodes which come to consensus on smart contract execution and use threshold signatures to execute the associated transactions on Bitcoin.
- Ark: Ark is building a novel layer 2 payments protocol for Bitcoin. The way it works is conceptually similar to the Lightning Network in that users lock funds in a UTXO and then transact offchain using VTXOs through transfers that are orchestrated through a central server, but without handing over custody to this centralized party.
- Babylon: Babylon is a Bitcoin staking solution which allows BTC holders to stake BTC on Bitcoin L1, taking on slashing risk, to provide economic security to proof-of-stake chains. BTC holders who stake can thus earn rewards and transaction fees from chains renting economic security.
- Lightspark (Lightning Network): Lightspark is building a platform to make the Lightning Network more accessible and useful, primarily for B2B payment use cases. Their platform aims to fix Lightning Network headaches through automated routing, liquidity management, compatibility with the traditional banking system and human-readable Universal Money Addresses.
- RGB: RGB is the best known client side validation protocol. It allows parties to transact off-chain in a privacy-preserving peer-to-peer manner, allowing smart contracts to be executed among these parties leveraging a Turing-complete programming environment called the AluVM.

Primer on Trust Assumptions

It's useful to think of trust in the context of layer 2 solutions as the number *m* of external parties that need to be trusted for a system to function as expected, out of a total of *n* parties involved. For example, when using a centralized custodian to safeguard funds you're trusting a single party, i.e., a 1 of 1 trust assumption. This can be fine if the party is trustworthy, but in a blockchain context the whole point is to remove trusted intermediaries and achieve trust among a decentralized set of unknown parties who may or may not be trustworthy. Stepping up a level, you could distribute trust across multiple parties by using a multi-signature (multisig) wallet, where for instance you only need 6 out of 10 possible singers to sign in order to move funds. You therefore only need to trust an honest majority of the parties, rather than the whole set. This is an improvement, since a single party no longer has complete control or needs to meet a high bar of trust.

The ideal trust assumption is 0 of N, that is, you don't need to trust any external parties and can verify everything yourself. The chart below visualizes the spectrum of trust assumptions, where the greener the better.

The Spectrum of Trust Assumptions



Source: vitalik.eth.limo, GSR.

About GSR

GSR has over a decade of extensive experience in the crypto market, serving as a trusted liquidity provider and active, multi-stage investor. Our suite of services includes OTC Trading, Derivatives, and Market Making. GSR is actively involved in every major sector of the digital asset ecosystem, working with token issuers, institutional investors, miners, and leading trading venues.

Find out more at www.gsr.io.

Follow GSR for more content: Twitter | Telegram | LinkedIn

Required Disclosures