OCTOBER 2024

UNDERSTANDING INTEROPERABILITY AND CRYPTO'S MULTICHAIN FUTURE

COMMISSIONED BY web3 foundation

THEBLOCK.CO

TABLE OF CONTENTS

SPONSORS & RESEARCH

SPONSORED BY web3 foundation

The Web3 Foundation's mission is to nurture cutting-edge applications for decentralized web software protocols. Their passion is delivering Web 3.0, a decentralized and fair internet where users control their own data, identity, and destiny.

RESEARCHED BY THE BLOCK PRO · RESEARCH

The Block Pro is The Block's premium product portfolio designed to help institutions evaluate opportunities in digital assets. Pro's research, news, and data products are powered by teams of subject matter experts deeply entrenched in the digital asset ecosystem who deliver actionable intelligence so businesses can make informed decisions.

The Block Research produces research content covering the digital assets, fintech, and financial services industries.

CONTACT

Email: research@theblock.co Twitter: @theblockres

ACKNOWLEDGMENTS

We would like to thank the Polkadot Foundation for commissioning this research report.

We would also like to thank everyone at The Block who assisted with this report, including Marcel Bluhm, George Calle, and Kevin Peng.

We are also grateful to those who shared their valuable perspectives through interviews for this report:

Sunny Agarwal - Osmosis

Arjun Chand - Li.finance

Filippo Franchini - Web3 Foundation

Sergey Gorbunov - InteropLabs (Axelar)

Michael Kaplan - AvaLabs

Seun Lanlege - Polytope Technologies

Dan Reecer (and team) - Wormhole Foundation

Kyle Samani - MultiCoin Capital

AUTHOR

Michael Thoma

The Block Pro Research Analyst

EXECUTIVE SUMMARY

Ever since Bitcoin spawned a new technological and financial revolution in 2008, the cryptocurrency space has not stopped growing, nor has its complexity. In today's world of hundreds of blockchains, achieving seamless communication and interoperability between networks can no longer be considered a "roadmap item"—it is a necessity for any protocol looking to garner meaningful adoption. This report explores how cross-chain interoperability is redefining the blockchain landscape of the past, allowing isolated ecosystems to transform into interconnected networks that can share data and value effortlessly.

At the heart of this transformation are bridges and interoperability solutions like Hyperbridge, LayerZero, Wormhole, and others that enable smart contracts to interact across numerous chains. This paradigm shift makes it possible to access decentralized finance (DeFi) protocols, liquidity pools, governance votes, and more on any chain. With over $80 billion currently locked in DeFi and hundreds of billions more expected in the coming years, a clear growth potential is unlocking the entirety of this TVL for all users across any chain.

Yet, cross-chain interoperability is a complex task, as each user, developer, and project may have differing priorities for their cross-chain experience. Users care about speed, low costs, and security, but achieving all three is a delicate balance. This report dives into what makes an ideal bridge, exploring key differences in mechanisms like burn and mint, general messaging protocols, and emerging trust-minimized solutions. As cross-chain interactions grow, understanding these nuances and minimizing the well-documented bridge hacks is crucial for legitimizing the blockchain space and onboarding the rest of the world to this new global, decentralized financial system.

Fortunately, cutting-edge technologies like zero-knowledge proofs (ZKPs), intent-based protocols, and aggregation layers are pushing the boundaries of what's possible across chains while simultaneously improving the user experience (UX) and overall security. Additionally, leading multi-chain ecosystems like Cosmos, Avalanche, and Polkadot are tackling interoperability in unique ways within their own growing ecosystems. Each offers a different path to a more interconnected blockchain world—whether through light client-enabled IBC connections in Cosmos or Polkadot's shared security model and Cross-Consensus Message Format (XCM) that enables frictionless parachains communication.

However, with these advancements in cross-chain interoperability come challenges, some new and some old. New code complexity, bleeding-edge technology, broader attack surfaces, oracle manipulation, and more are just a few obstacles new interoperability solutions must confront.

This report is your guide to understanding how these emerging solutions are not just connecting blockchains but reshaping the future of the entire crypto ecosystem. Explore the elaborate mechanics, learn from real-world examples, and see what's next in the race toward a more unified, scalable, and secure blockchain universe.

PART 1
OVERVIEW & KEY CONCEPTS

As blockchain technology evolves, the need for seamless communication across distinct networks has become increasingly critical. Cross-chain interoperability addresses one of the blockchain ecosystem's fundamental challenges: fragmentation. Historically, blockchains like Ethereum, Bitcoin, and others were built as isolated ecosystems, limiting their interaction and constraining the potential of decentralized finance (DeFi) and other blockchain applications. This report delves into interoperability's critical role in overcoming these barriers, creating scalable and efficient ecosystems where value and information can move freely. However, before diving right into the complex world of blockchain interoperability, let's take a moment to discuss a few fundamental concepts and definitions.

1.1 CROSS-CHAIN INTEROPERABILITY

Cross-chain interoperability refers to the ability of distinct and disparate blockchain networks (e.g., Polkadot and Ethereum) to communicate, share data, and exchange value. Unfortunately, from the beginning of blockchain experimentation, each blockchain has been designed as an isolated ecosystem unaware of other chains and unable to communicate with anything outside of its protocol. A blockchain, by its definition, is a shared, immutable ledger that records every transaction posted by its users. For two separate blockchains to interact, they would need to agree on a common state and maintain an immutable record of every transaction on each other's networks. The volume of data that would need to be exchanged and stored to achieve this for multiple blockchain pairs makes direct interaction impractical and challenging to scale. Extending this model to every pair of blockchains wishing to communicate only exacerbates the complexity.

> "Blockchains are coordination mechanisms, and it doesn't make sense that they should be siloed."
> -Seun Langlege, Polytope Solutions

Cross-chain interoperability solves these problems by allowing different blockchains to exchange data and value without each having to manage the other's entire state directly. It effectively bridges two blockchain networks, removing the need for a third-party intermediary like a centralized exchange (CEX). This bridging function is key, allowing different blockchain systems to be part of a larger interconnected ecosystem rather than isolated entities.

In practice, bridges enhance the scalability of blockchain networks by eliminating the limitations imposed by isolated environments. Just as interoperability between different cellular network carriers allows users to communicate across networks, cross-chain bridges enable blockchain networks to interact seamlessly. This interaction is crucial as the blockchain ecosystem continues to expand, with new rollups and appchains being introduced regularly.

Key use cases for cross-chain interoperability are,

1. **Collateral**: Blockchains like Bitcoin store immense value but contain little to no DeFi ecosystem. Therefore, users get very little utility out of this capital. Cross-chain bridges could unlock DeFi for assets on these blockchains and allow for more efficient use of capital.

2. **Scalability**: Many blockchains have limited scalability and suffer during periods of increased transaction volume. Cross-chain interoperability can reduce the frictions of switching blockchains, enabling users to move to another chain should their current chain suffer from congestion or high fees.

3. **Efficiency**: Along with scalability, users may be willing to conduct some transactions/activities on a chain that, while having less economic security than Ethereum, offers cheaper fees and faster finality (like Polygon). Similarly, competing L1 blockchains and dApps may offer higher yields to attract users and liquidity. Users can bridge their assets from one chain to another to access these higher yields.

4. **Web3 Adoption**: Web3's success relies heavily on blockchain interoperability. Users will want to keep their avatars, currencies, and other assets across different games and experiences, and cross-chain bridges will allow that.

5. **Wrapping or Unwrapping Native Assets**: To get a native token on its original chain, a user may need to bridge a wrapped (a bridged, IOU version of a token not on its native blockchain). This allows them to move their assets from one chain to another and take possession of the native token directly on the destination chain.

To understand the impact and potential of cross-chain bridges, we must look at their place in the bigger picture. As of 2024, the total value locked (TVL) in cross-chain bridges is around $6 billion. While this is a significant sum at face value, it's only a small fraction of the ~$86 billion of assets locked in DeFi protocols.

Being able to source and transfer liquidity across the entire blockchain ecosystem would increase the liquidity of most blockchains by orders of magnitude. It would make DeFi protocols more efficient and create a more connected and robust blockchain world where assets and data can move freely across networks, allowing for more innovation and more opportunities for users and developers.

1.2 COMPOSABILITY

Interoperability and composability are critical terms in the blockchain space that describe how different parts of apps and chains interact. As previously described, interoperability permits moving assets and messages between various apps and blockchain networks. It allows for data and value to flow across different ecosystems.

On the other hand, composability is about building a shared infrastructure between apps so developers can deploy and interact with multiple apps across multiple chains with minimal effort. This means an application can be built once and then deployed across multiple different blockchains. In addition, the application can easily be built upon and/or incorporated into other applications with no additional developer work. This ability inspired the common DeFi descriptor, "money legos," because builders can build new primitives simply by using and/or combining already existing applications. Composability is particularly valuable because it offers users more choices and flexibility, enabling them to perform actions across numerous ecosystems without the need to build everything from scratch.

The DeFi space is worth around $86 billion and is dominated by Ethereum and its Ethereum Virtual Machine (EVM). In a world where interoperability doesn't exist, users on non-Ethereum chains like Solana or Polkadot cannot interact with the value being generated in the Ethereum-dominated DeFi space. In that state, each DeFi ecosystem can be likened to a separate economy; however, these economies would struggle to grow effectively if they could not interact.

Cross-chain interoperability can increase the adoption of DeFi by allowing users across different blockchain networks to access DeFi protocols seamlessly. This increased accessibility gives users more value from DeFi, as the limitations of a single chain no longer constrain them. As accessibility improves, more users will be drawn to Web3 and DeFi, and more liquidity will flow into the DeFi space. This will enable a larger scale for activities like lending, staking, yield farming, and borrowing and strengthen the DeFi sector.

1.3 GENERAL MESSAGE PASSING

Recent advancements in blockchain interoperability have moved beyond token transfers to more complex and flexible solutions like general messaging. Protocols like LayerZero, Wormhole, and the upcoming Hyperbridge are examples of this trend. These three, and dozens of others in the space, open up new possibilities for dApps by allowing them to interact seamlessly across multiple blockchains without requiring direct deployment on each one. These new advanced use cases include cross-chain lending, arbitrage, governance, and more that have previously been impossible, impractical, or uneconomical to execute.

When two blockchains can communicate via generalized messaging, dApps gain the ability to make contract calls across chains, vastly expanding their utility beyond simple token transfers. General Message Passing (GMP) allows developers to create interchain-native applications by enabling cross-chain function calls and seamless state synchronization, abstracting the complexity away from users. Unlike traditional bridging, which relies on wrapped assets, GMP transmits data and function calls directly, reducing gas fees and minimizing asset fragmentation. This streamlined approach simplifies provenance tracking by keeping assets on their original chains and supports smoother, single-click user experiences, driving broader adoption of decentralized applications.

In the near future, users can expect several significant benefits from the development of cross-chain smart contracts and generalized messaging protocols:

- **Super Wallets**: These advanced wallets will enable users to control sub-accounts across various chains from a single address, simplifying management and improving user experience.
- **Aggregated DeFi Services**: Cross-chain yield farming, collateralized loans, and other DeFi services will become more accessible and efficient, leveraging the liquidity and functionality of multiple blockchain networks.
- **Unified Governance Systems**: Cross-chain DAOs (Decentralized Autonomous Organizations) will allow for seamless governance across multiple blockchains, creating more cohesive and integrated systems.

PART 2
WHAT MAKES FOR AN "IDEAL" BRIDGE?

"Bridges have a few major ways to differentiate:
- Speed (how long does it take to bridge)
- Slippage costs
- Gas cost
- Trust-properties

Most users tend to prioritize the first 3; most Ethereum researchers tend to focus on the 4th."
-Kyle Samani, MultiCoin Capital

Each blockchain bridge may have specific criteria for which they optimize, but at a high level, nearly all bridges strive to achieve three primary characteristics:

1. **Security/Trust-minimization**: The most crucial aspect of any bridge is the guarantee that information and assets can be securely verified and transferred across chains without the need for trusted third parties.

2. **Superior UX**: The primary factors that determine a good user experience include low costs, constant uptime (liveness), numerous cross-chain connections, and fast finality/settlement. Additionally, a bridge that offers more liquidity provides a better user experience by reducing slippage and ensuring better pricing.

3. **Native Assets**: Users prefer native assets over wrapped alternatives due to the latter's lower liquidity and security guarantees.

Native assets on blockchain networks are tokens issued directly at the protocol level, like ETH on Ethereum or SOL on Solana. These assets are tightly integrated with their respective blockchains and managed on-chain without third-party involvement. However, the situation is more complex for stablecoins like USDC, which can be minted on multiple blockchains but involve nuanced management across different networks.

Native assets offer a more seamless and reliable user experience. In contrast, bridged assets represent a fragmented class of tokens, acting as IOUs backed by equivalent assets on another chain. When a stablecoin like USDC is bridged, the original asset is locked on the source chain, and a new token is minted on the destination chain. This process creates fragmentation as different organizations mint their own versions of bridged tokens, leading to a lack of standardization and coordination.

Wrapped assets, a type of bridged asset, are generally considered inferior to native assets due to several risks. These include counterparty risk, liquidity fragmentation, depegging risks in DeFi protocols, and reliance on external oracles for pricing data, making them less reliable and more complex than native assets.

> "You can have the most secure bridge in the world, but if the asset you end up with has no liquidity, it is not very useful for the end user.
> -Michael Kaplan - AvaLabs

In addition to these core characteristics, other factors play a significant role in evaluating a bridge's performance:

- **Connectivity and Asset Support**: How many networks can the bridge connect to? How generally extensible is the technology? Some bridges, like Axelar, aim to connect nearly all chains and VMs, whereas others, due to their design, are more narrowly focused, like the Hop bridge that mainly connects Ethereum with its rollups. Similar to the number of blockchains a bridge can support, the more assets that can be bridged, the better the UX.

- **Diverse Validator Set (if applicable)**: The validator set overseeing the bridge must consist of well-established, experienced entities that are independent, diverse, and consistently monitored. These validators should operate with clear cryptoeconomic incentives and be auditable by the public to ensure both their performance and security track record. Admin privileges should be distributed carefully, ensuring accountability and transparency in governance decisions, while key protocol operations and potential anomalies must be publicly auditable and monitored. Decentralized stake and stable token dynamics further solidify the integrity of the bridge's operations.

- **Mature Implementation**: The protocol should be open-source and extensively tested, with thorough audits backing its reliability. Detailed and transparent technical documentation is critical, offering insight into the protocol's design and security features. Additionally, well-defined emergency response mechanisms and governance procedures are essential for handling unexpected issues or upgrades.

- **Extractable Value: Cross-chain Maximum Extractable Value (MEV)** introduces complexity by leveraging multiple chains, assets, and protocols, raising concerns about the potential for limitless strategies beyond single-chain MEV. What is the risk of intermediaries extracting value from transactions (e.g., MEV from Flashbots)?

PART 3
CURRENT BRIDGING & INTEROPERABILITY LANDSCAPE

The bridging and interoperability space in DeFi has seen significant changes and evolution, especially from 2022 through 2024. Below is a look at some of the top protocols in this space, ranked by one-month volumes (Figure 1), as well as quarterly volumes for the entire blockchain space over the course of 2024 (Figure 2).

Figure 3.1. Top 10 Bridges, Based on 30-Day Volume

	Name	Chains	1d Change	24h Volume	7d Volume	1m Volume
1	Stargate	+10	-44.27%	$26.2m	$256.94m	$1.375b
2	Circle CCTP		-25.32%	$30.99m	$225.09m	$1.145b
3	IBC	+86	-16.64%	$16.3m	$130.63m	$875.82m
4	Across	+4	-34.29%	$20.44m	$155.95m	$719.26m
5	Portal by Wormhole	+7	-55.71%	$8.05m	$86.4m	$541.54m
6	deBridge	+4	-28.55%	$7.41m	$75.97m	$349m
7	Orbiter Finance	+10	+8.17%	$12.71m	$115.91m	$340.44m
8	Arbitrum Bridge		-51.83%	$5.2m	$87.52m	$338.48m
9	zkBridge	+7	+24.26%	$13.26m	$66.14m	$292.38m
10	Synapse	+9	-14.32%	$5.65m	$34.56m	$218.4m

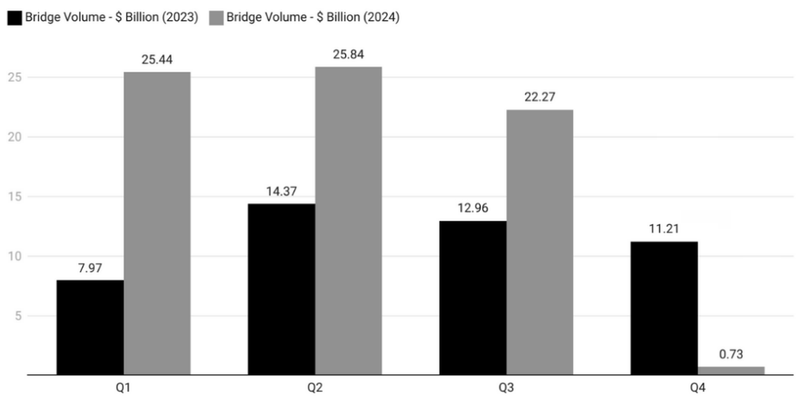
Source: DefiLlama

THE BLOCK PRO · RESEARCH

Source

The current bridging and interoperability landscape in 2024 is led by a mix of protocols that have carved out unique roles in the ecosystem. Stargate and LayerZero have seen notable adoption, with Stargate excelling in cross-chain liquidity provision and LayerZero gaining traction through its developer-friendly focus on cross-chain messaging. Axelar has emerged as a strong contender with a surge in user base, offering general message passing that extends beyond asset transfers, while Wormhole maintains steady usage for bridging assets between major chains like Ethereum and Solana.

Figure 2. Quarterly Bridge Volume, 2023 vs. 2024



Source

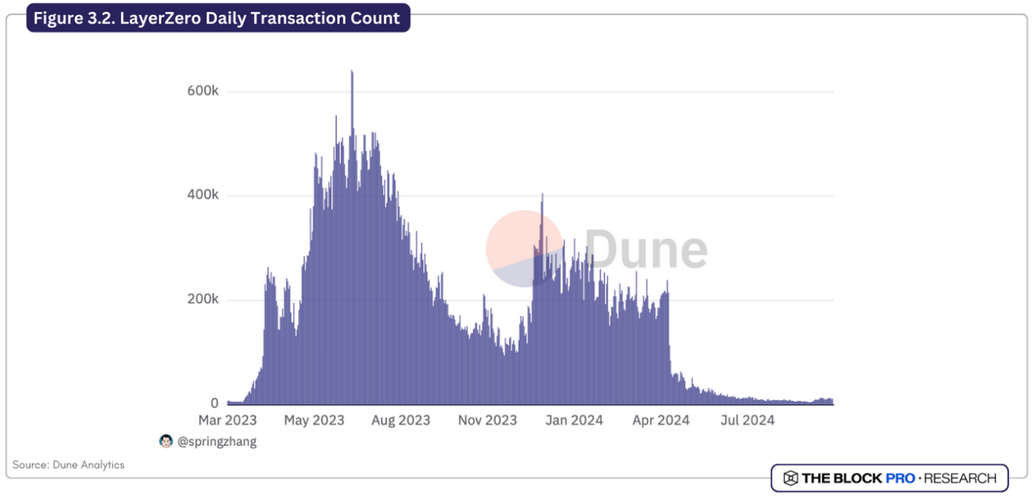
Across and Orbiter are gaining popularity for their focus on fast, low-cost transfers between Ethereum Layer 2 networks, appealing to users seeking efficient asset movement. Meanwhile, IBC (Inter-Blockchain Communication) remains crucial within the Cosmos ecosystem, facilitating seamless communication and interoperability between its various blockchains. These protocols collectively address different needs, from high liquidity requirements to quick L2 transactions, shaping the next phase of DeFi's cross-chain interactions.

Key Trends in Bridging Protocols consist of

- **Security Concerns**: Bridge hacks accounted for a large portion of DeFi losses in 2022, forcing the industry to reevaluate bridging security. While not perfect, the total amount of funds lost via bridge hacks in 2024 is declining compared to 2022.
- **A Shift to Layer 2**: Many bridging protocols are integrating more deeply with Ethereum Layer 2s like Arbitrum and Optimism. Dozens of Ethereum rollups have launched in the last ~two years, forcing many general bridging solutions to integrate them into their protocol.

- **Ecosystem Growth**: Axelar and Across have seen a rise in adoption, particularly in 2024, as they have positioned themselves as more general interoperability and intent-based solutions. LayerZero, while still an industry leader, has seen activity decline dramatically (Figure 3) after the launch of its token, ZRO, suggesting that much of the 2023 and early 2024 activity was due to airdrop farming.

Figure 3.2. LayerZero Daily Transaction Count



Source

As illustrated below in Figure 4, users have plenty of bridge options but also face drawbacks. Every design choice, whether it is upgradeable smart contracts, third-party validators, or something else, introduces a new security trust assumption or attack vector.Unfortunately, many casual crypto users are not well-enough informed to understand the technical differences and risks associated with each bridge. The crypto space lacks a "perfect" solution, and as we cover in this report, one may not exist for the entirety of the space.

Figure 3.3. An overview of the different security designs and mechanisms in the leading token bridges

#	NAME	DESTINATION	VALIDATED BY	TYPE	SOURCE UPGRADEABILITY	DESTINATION TOKEN
1	Polygon PoS	Polygon	Destination Chain	Token Bridge	Yes	Canonical or Wrapped
2	LayerZero v2 OFTs	Various	Third Party	Token Bridge	Yes	Canonical
3	Polygon "Plasma"	Polygon	Destination Chain	Token Bridge	Yes	Native & Canonical
4	Ronin V3	Axie Infinity Chain	Third Party	Token Bridge	Yes	Canonical
6	Portal (Wormhole)	Various	Third Party	Token Bridge	Yes	Canonical or Wrapped
7	Omnichain (LayerZero)	Various	Third Party	Token Bridge	Yes	Canonical
8	Omnibridge	Gnosis Chain	Third Party	Token Bridge	Yes	Canonical
11	Avalanche Bridge	Avalanche	Third Party	Token Bridge	EOA	Wrapped
12	Rainbow Bridge	Various	Destination Chain	Token Bridge	Yes	Canonical or Wrapped

Source: L2Beat

THE BLOCK PRO · RESEARCH

Source

PART 4
BRIDGING MECHANISMS FOR TOKEN TRANSFERS

This section explores five key methods for facilitating cross-chain interoperability: burn-and-mint, lock-and-mint, liquidity pools, atomic swaps, and omnichain decentralized exchanges (DEXs). Each approach has unique trade-offs in security, decentralization, liquidity management, and user experience, shaping the strategic considerations for developers, users, and liquidity providers alike.

From centralized bridges relying on validators to decentralized liquidity pools and atomic swaps, each mechanism aims to safely and efficiently address the inherent challenges of moving assets between blockchains. Innovative solutions like AI-driven liquidity management and general message passing further enhance interoperability as the blockchain ecosystem evolves. Understanding these models and their implications is essential to building a more connected, scalable, and secure blockchain ecosystem that enables users to fully realize the potential of Web3 and DeFi across multiple networks.

4.1 BURN AND MINT

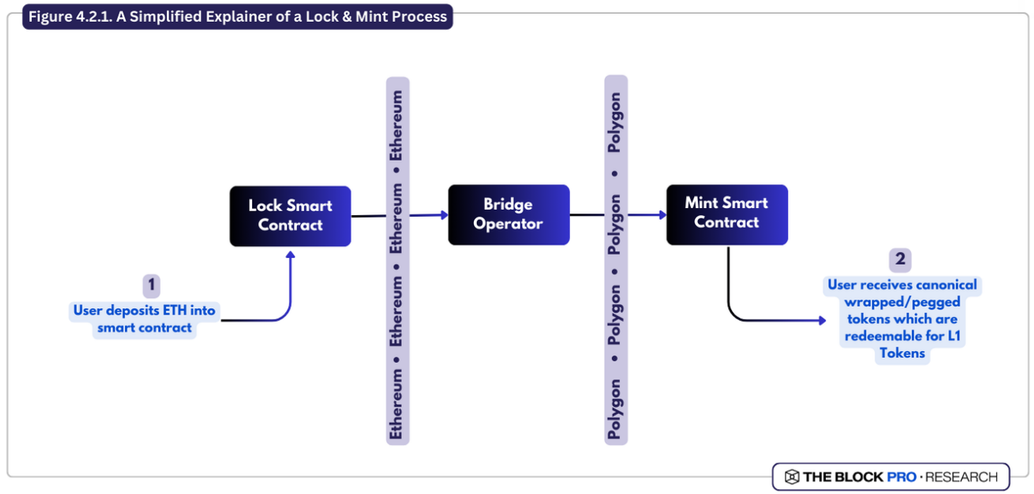
Some bridges use off-chain validators and a burn and mint mechanism to enable bridging between different L1s. Validators, in this scenario, are third-party entities that manage the bridged assets and the overall functionality of the bridge. This introduces new trust assumptions specific to the third-party validator set and outside the security guarantees of the L1s.

To simplify the complexities of cross-chain interoperability, one approach is to centralize trust in a single cross-chain entity: the bridge entity. Instead of managing canonical assets on a per-chain basis, the focus shifts to adopting a per-bridge canonical asset. These assets are created and maintained using a bridge's proprietary cross-chain token standard, such as LayerZero's Omnichain Fungible Token (OFT), Wormhole Native Token Transfer, or Axelar Interchain Token Service. Like the ERC20 token standard, these standards facilitate the minting and burning of bridged assets on destination chains.

In the burn and mint model, the bridge burns (permanently destroys) tokens on the source chain while simultaneously minting an equivalent number of tokens on the destination chain (Figure 5), each representing the transferred asset on its respective chain. The critical aspect is that both actions—burning and minting—must occur simultaneously. Examples of this design include Hop Protocol and any L2-L2 bridges.

This "burn and mint" mechanism is akin to a "lock and mint" process (explained further below), with the key difference being that tokens are burned on the destination chain rather than simply "locked," which implies they can re-enter circulation at a later date. This approach ensures that the source chain remains the definitive source of truth, simplifying cross-chain token management by providing a unified view of the tokens in circulation.

Figure 4.2.1. A Simplified Explainer of a Lock & Mint Process



Source

However, this convenience comes with the drawback of vendor lock-in. Per-chain canonical assets lack fungibility across different bridge providers, meaning switching providers is difficult and costly. Token issuers are limited to the chains supported by their chosen bridge provider and are subject to the provider's fee structures, such as cross-chain token minting fees. This dependence on a single provider underscores the challenges and strategic considerations of adopting a per-bridge canonical asset standard.

4.2 LOCK AND MINT

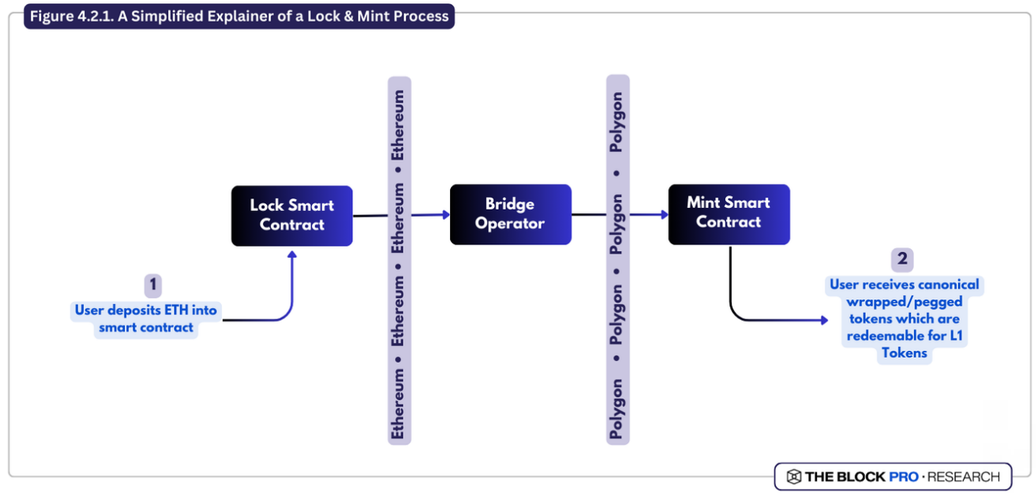
In a lock and mint process, assets are locked on the source chain, and an equivalent synthetic or wrapped token is minted on the destination chain (Figure 6). Locking tokens involves restricting their transferability without destroying them, meaning they technically still exist but are effectively removed from circulation

from a financial standpoint, as the total supply remains unchanged. However, from a technical perspective, these locked tokens could still be transferred if the bridge logic is flawed, posing a risk not found in the burn and mint approach.

With this design, native tokens are generally locked on the source chain, and wrapped tokens are minted on the destination chain. The security of this system heavily relies on the bridge and its network of validators, making it somewhat centralized and vulnerable. For example, Wrapped Bitcoin (wBTC) on Ethereum, which is secured by just a few centralized entities, holds ~$9 billion in value locked up in the bridge.

Smart contracts deployed on each chain can interact through the bridge, enabling the transfer of assets between systems. This is done through two smart contracts: the Vault contract, which holds the assets on chain A, and the Issuance contract, which can issue an IOU on chain B. The deposit process is when a user deposits coins into the Vault on chain A, and the bridge issues the same amount of assets to the user on chain B.

Figure 4.2.1. A Simplified Explainer of a Lock & Mint Process



Source

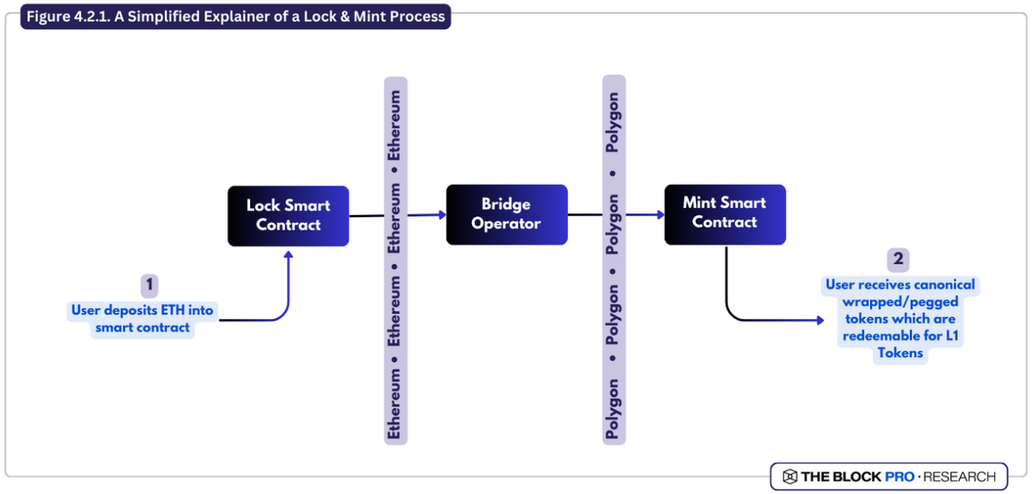
In this scenario, the safety of your assets relies entirely on the bridge contract and the external validator committee. If the bridge is compromised, your assets can be instantly stolen. For example, when you lock ten ETH on the Ethereum side of an Ethereum-Solana bridge and receive ten wETH on Solana, the value

of that ten wETH is based on the expectation that it can be exchanged back for ten ETH via the bridge. However, if an exploit occurs and the ten ETH locked on Ethereum is stolen, the ten wETH on Solana loses its backing and becomes worthless. Since these transactions are valid on both chains, reversing the theft is impossible.

4.3 CROSS-CHAIN LIQUIDITY POOLS

Liquidity bridges, such as Hop and ChainPort, connect value between blockchains using liquidity pools where Liquidity Providers (LPs) act as middlemen. LPs create and maintain liquidity pools on both the source and destination chains to ensure adequate liquidity for transactions between the two chains (Figure 7). For this service, the LPs typically get paid by charging a fee. In cross-chain swaps via liquidity pools, no assets move between chains. Instead, assets are "traded" between chains. For example, if you have funds on Chain A and need them on Chain B, an LP with the same asset on Chain B will swap it for your asset on Chain A and charge a fee for the service.

Figure 4.2.1. A Simplified Explainer of a Lock & Mint Process



Source

The liquidity pool design utilizes the validator sets of the underlying chains during cross-chain swaps. However, in most cases, only two validators - one from each chain - verify the counterparty on the other chain instead of the entire validator sets of both chains.

Validators in liquidity networks act as "routers" that manage liquidity pools, verify the counterparties, and facilitate atomic swaps. "Atomicity" (discussed in the next section) is a software term that refers to database transactions that either execute in full or not at all. In the context of blockchain transactions, an atomic transaction either executes every aspect of the multi-step transaction or reverts completely, avoiding any partial results. Most of the risks in liquidity bridges are with the LPs who have their funds at risk in the pools, while bridge users are mainly concerned with speed and liquidity. For LPs, volume and security are key as they directly impact how they earn fees and protect their assets.

This system functions effectively when it is well-capitalized, decentralized, efficiently rebalanced, and incentivizes LPs to maintain ample liquidity on both sides of the bridge. Balancing all of these aspects across an ever-growing number of blockchains is quite challenging. However, two innovative approaches to liquidity management that have been growing in adoption are from Across and Stargate. Across uses a unified liquidity pool on Ethereum, consolidating most of its assets into one location to avoid maintaining separate pools across networks. Liquidity is transferred only when needed, ensuring quick responses, preventing shortages, and minimizing fees. Similarly, Stargate's V2 protocol introduces an AI-driven Planning Module (AIPM) that monitors liquidity across networks in real-time. AIPM dynamically adjusts fees and rewards to incentivize balanced liquidity distribution, optimizing the flow of assets between chains.

4.4 ATOMIC SWAPS AND HTLCS

Atomic swaps allow two parties to directly exchange one cryptocurrency for another, such as ETH for BTC, without relying on a third party. The term "atomic" refers to the all-or-nothing nature of the trade, ensuring that neither party can walk away with all the tokens while the other gets nothing. This is the primary benefit of atomic swaps, which are often touted by the bridges that utilize them, such as SafeSwap. These swaps are often implemented using Hashed Time Lock Contracts (HTLCs), which are time-sensitive, condition-based agreements that guarantee the trustless exchange of assets.

An HTLC acts as a secure, temporary escrow, holding funds until both parties fulfill their obligations. The contract uses a cryptographic hash and a private key to control access to the funds, requiring each party to submit cryptographic proof that they have met their side of the deal. If these proofs aren't provided within a pre-set timeframe, the deposited tokens revert to their original owners.

For instance, if Alice wants to swap 10 X tokens with Bob for 10 Y tokens, they would create an HTLC that expires in one hour. Alice deposits her tokens into a contract address and shares a cryptographic hash with Bob, who verifies the deposit and then deposits his tokens into a similar contract. Alice can then claim Bob's tokens using her private key, which also allows Bob to claim Alice's tokens, completing the swap.

Despite their advantages, atomic swaps are not without challenges. They can be complex and time-consuming, particularly for beginners, requiring both blockchains to use the same hashing algorithm.

4.5 CROSS-CHAIN OMNIDEXES

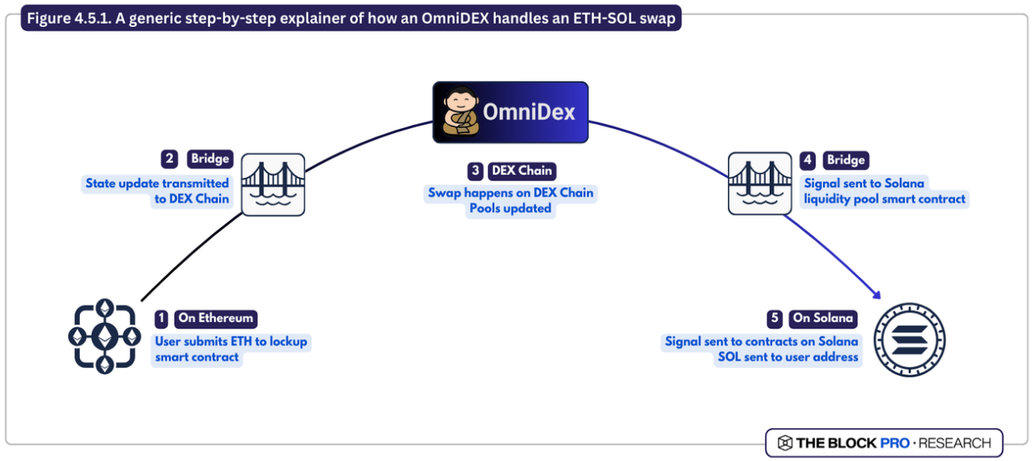
An Omnichain Decentralized Exchange (DEX) enables liquidity providers to supply native liquidity across multiple blockchains. Some omnichain DEXes use proprietary tokens such as THORChain's RUNE and Osmosis' OSMO (sometimes) to bridge liquidity between chains (Figure 8). The protocol typically performs two swaps (e.g., ETH to RUNE to SOL) to facilitate the exchange of long-tail assets across chains. Additionally, some Omnichain DEXes may launch a dedicated blockchain specifically for handling DEX operations.

THORChain manages the swapping process through secure and transparent smart contracts that guarantee the irreversibility and integrity of each swap. To ensure smooth transactions, THORChain maintains liquidity pools that provide the necessary assets for swapping. For example, if you want to exchange BTC for ETH, THORChain checks the liquidity pool to confirm there is enough ETH available. During the swap, your assets are securely stored in vaults, which protect them and ensure the transaction's accuracy. Once the swap is complete, the newly acquired assets, such as ETH, are immediately transferred to your designated wallet address.

The steps in a THORChain cross-chain swap are:

1. The user sends BTC into THORChain.
2. Once the BTC is received by the THORChain protocol, RUNE is transferred from the BTC pool to the ETH pool, effectively performing a double swap—first BTC to RUNE, then RUNE to ETH.
3. The ETH is then released from one of THORChain's vaults to the user.
4. In this scenario, THORChain functions as the external validator system that effectively manages vaults across connected blockchains. In this role, it acts as the intermediary between the chains being bridged. Therefore, the cryptoeconomic security of the entire THORChain network serves as the central trust assumption for its operation. The security and integrity of the network rely on the collective stake and economic incentives within THORChain, so it operates securely as it coordinates asset exchanges and transactions between chains.
5. At the heart of THORChain is a state chain that manages asset and exchange logic while delegating transaction output to specific endpoints on each connected blockchain. These endpoints handle transactions for their respective chains. Each signer in this system needs to run a full node on the connected blockchains to have a complete and up-to-date view of the blockchain's state. The clients on each chain are relatively lightweight, containing only the necessary logic to interact with contracts on that specific chain. The heavy lifting is done by observer clients that operate on THORChain itself. These observer clients monitor and coordinate the overall network activity, ensuring that transactions are executed correctly and securely across the connected blockchains.

Omnichain DEXes are more permissionless than CEXs as they do not require KYC. They also have smart contract composability and unified liquidity where all chains draw from the same liquidity pool (e.g., RUNE-SOL). This means native assets are swapped, and there is no need to trust the DEX once the transaction is complete.



Source

However, Omnichain DEXes have some limitations. One issue is the lack of immediate finality due to potential concurrent calls from multiple chains to the same liquidity pool, which can lead to complications in transaction execution and potential reversion or refunding. Another concern is the reliance on a middle chain, which represents a single point of failure. Additionally, users may face multiple layers of fees and slippage because DEXes like THORChain and Osmosis require native tokens (e.g., THOR) to facilitate the swap. Finally, like Automated Market Maker (AMM) DEXes, Omnichain DEXes suffer from low capital efficiency, which remains a significant challenge.

PART 5
GENERAL MESSAGING PROTOCOLS

Protocols like LayerZero, Wormhole, Axelar, and others represent a significant advancement in cross-chain communication, enabling seamless interaction between smart contracts operating across distinct blockchain networks. In essence, these projects facilitate the transfer of messages from a source network to a destination network, where they are executed according to predetermined protocols. This innovative technology underpins the development of omnichain applications (OApps) by allowing them to connect with multiple blockchains, thereby extending their functionality and reach.

5.1 LAYERZERO

LayerZero originally used a combination of relayers and oracles to ensure secure and efficient message passing across chains. However, LayerZero V2 uses an "intrinsic security" design that reduces the number of security assumptions and ensures messages are delivered without loss or duplication.

LayerZero V2 uses a "mesh" network framework that aims to develop a universal communications standard/language that applies to all connected chains. This allows for minimal overhead when adding dozens of new chains. A key feature of V2 is the shift from the traditional Oracle and Relayer model to a more advanced system of Decentralized Verification Networks (DVNs) and permissionless Executors. DVNs are responsible for verifying the message before it's executed on the destination chain. Unlike the previous model, DVNs operate in a permissionless environment. Anyone from any external network, organization, or application can participate in the verification process. Currently, there are ~30 active DVNs, including Google Cloud, Animoca, and BlockDaemon.

In V2, Executors replaced Relayers and inherited their responsibilities of delivering and executing messages on the destination networks. The permissionless Executors, which are separate from the verification layer, improve operational speed and reduce the risk of bottlenecks. By decoupling execution from verification, LayerZero minimizes the chance of failure due to a single point of failure in the network. Executors are incentivized by a portion of the fees users pay for sending messages to ensure the network is robust and reliable.

All of LayerZero's V2 aspects can be divided into two layers: an immutable execution layer with consistent functionality across different chains and a configurable verification layer. The permissionless execution

model allows anyone to execute verified messages, not just a small group of pre-approved validators like in some bridging designs. Applications can customize their security by choosing DVNs and have tailored security and fee structures.

However, reliance on just a few DVNs poses risks, including potential single points of failure. Additionally, LayerZero's upgradability, while offering flexibility, raises concerns about security and decentralization, as multisigs can make changes to key parameters. However, this is not unique to LayerZero, as most interoperability protocols and bridges have some sort of upgradeability/multisig.

5.2 AXELAR

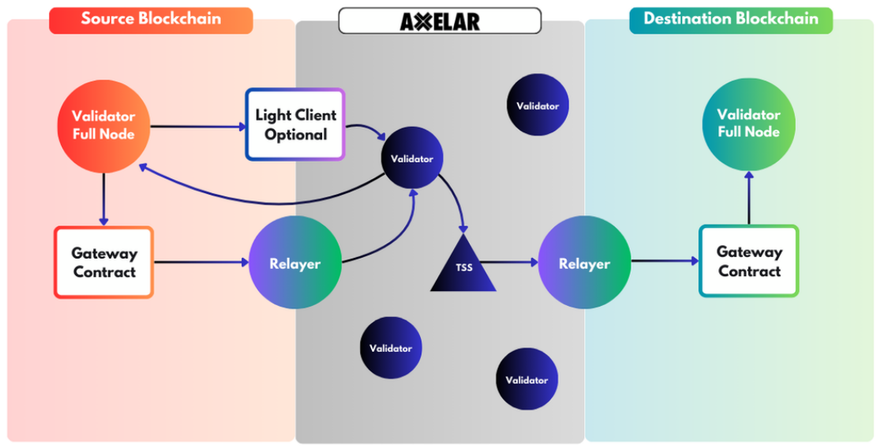
Axelar serves as a middleware that facilitates the transmission and verification of cross-chain transactions through appchains built with the Cosmos SDK. Utilizing the Tendermint consensus algorithm and a unique message verification mechanism, Axelar integrates seamlessly within the Cosmos ecosystem and supports inter-chain messaging for 60+ EVM chains.

Axelar operates with a Delegated Proof-of-Stake (DPoS) consensus mechanism, where validators produce blocks and manage inter-chain transactions. To prevent the concentration of voting power, Axelar implements quadratic voting, requiring validators to increase their stake exponentially to gain additional voting influence.

The Axelar Virtual Machine (AVM) allows for the permissionless connection of new blockchains and supports the deployment of smart contracts. Axelar's ecosystem includes two key components: a decentralized network of validators based on Cosmos and gateway smart contracts for connectivity between the Axelar network and external chains (Figure 9). Light clients can also be utilized if the chains involved utilize them. The use of a gateway contract for cross-chain messaging relies on sharded keys based on the stakes of validators on the Axelar hub. When a message is received, validators convert their verification results into votes, ensuring scalability and decentralization.

One final key AVM-powered service is the Interchain Token Service (ITS), designed for seamless token transfers across multiple blockchains while maintaining token fungibility and custom features. Unlike other solutions, Axelar's ITS supports standardized tokens and offers native support for data transfers.

Figure 5.2.1. Axelar Network Cross-Chain Message Communication



Source

5.3 HYPERBRIDGE

Polkadot's XCM and XCMP protocols (discussed in later sections) work well when interacting within Polkadot's shared security environment. However, when blockchains outside this shared security interact, a new solution is needed.

Hyperbridge, developed by Polytope Labs and launched as a parachain within the Polkadot ecosystem, is a cross-chain solution designed to address this exact need. It extends interoperability beyond Polkadot's ecosystem using specialized bridging contracts. These contracts translate external blockchain transactions into a format compatible with Polkadot's environment, enabling seamless interactions between Polkadot-based chains and other blockchains.

Hyperbridge functions as an "interoperability coprocessor," enabling secure communication and transactions across various blockchain networks outside of the Polkadot ecosystem. Operating as a

standalone parachain on Polkadot, Hyperbridge facilitates connections with Ethereum rollups such as Arbitrum, Base, Optimism, and other ecosystems like Binance Smart Chain. This setup enables native asset transfers and cross-chain messaging with a focus on minimizing hacking risks.

In addition, most computations within the Hyperbridge occur off-chain, and the outcomes, along with cryptographic proofs, are subsequently verified on-chain. By using Polkadot's BEEFY consensus proofs, Hyperbridge validates the transitions of different parachains, allowing it to distribute computational workloads across various Parachain Cores. A decentralized network of relayers manages asset transfers, applying cryptographic proofs to maintain the protocol's integrity. In addition to utilizing offchain compute and onchain proofs, Hyperbridge integrates an Interoperable State Machine Protocol (ISMP) feature to enable general message passing. ISMP supports not just asset transfers but also the execution of complex logic across different blockchain networks.

Hyperbridge directly benefits from Polkadot's shared security and governance, ensuring that transactions processed through Hyperbridge receive the same level of security and finality as any native Polkadot transaction. Hyperbridge capitalizes on Polkadot's consensus layer to validate the state and transitions of multiple chains, ensuring consistent security across the network.

PART 6
VERIFICATION DESIGNS & MECHANISMS

Interoperability in blockchain systems is fundamentally about facilitating communication between siloed blockchains and who/what the user is willing to trust to accomplish this. Every interoperability solution aims to minimize the amount of trust needed to interact across ecosystems. With different bridge designs, trust can be placed in the chain, the bridge and its validators, or with the issuer, who has ultimate control over the assets and their usage.

The pinnacle of trust minimization is a solution that inherits the trust assumptions of the most secure chain involved in the transaction, like Ethereum rollup native bridges. This design, where there are no added risks or trust assumptions beyond the chain(s) involved, is often called a “trustless” bridge. This is because if a user is holding ETH on the Ethereum mainnet, they are already comfortable with the security guarantees provided by Ethereum. However, they ideally do not want any additional risk when using a bridge or interoperability solution.

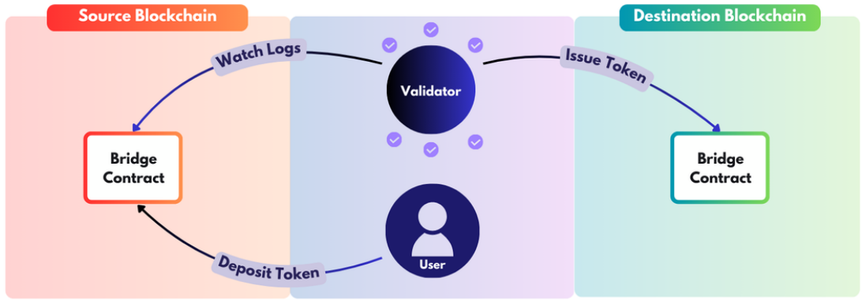
6.1 EXTERNAL VERIFICATION

External verification in blockchain bridges (Figure 10) involves using external validators, or relayers, to transfer messages and assets between two chains. This process introduces an additional layer of trust outside of the two blockchains involved, as the validators are responsible for securing the bridge and ensuring the transactions are processed correctly. Once a consensus is reached among these validators regarding the initial bridging request, the transaction is finalized on the destination chain after both chains approve it.

This approach is popular due to its relatively simple implementation. However, it comes with significant trade-offs, particularly regarding security, decentralization, and resistance to censorship. The reliance on a small group of validators introduces the risk of collusion or error, as the security of the transaction no longer depends solely on the underlying blockchains but also on the integrity of these external parties.

Key considerations for externally verified bridges include determining who has access to the locked assets, how information is relayed between chains, how the bridge verifies the accuracy of this information, the incentives provided to relayers, and who has the authority to trigger the minting and burning of tokens. These factors are essential to ensure the bridge's security and functionality.

Figure 6.1.1. An externally validated bridge explainer



Source

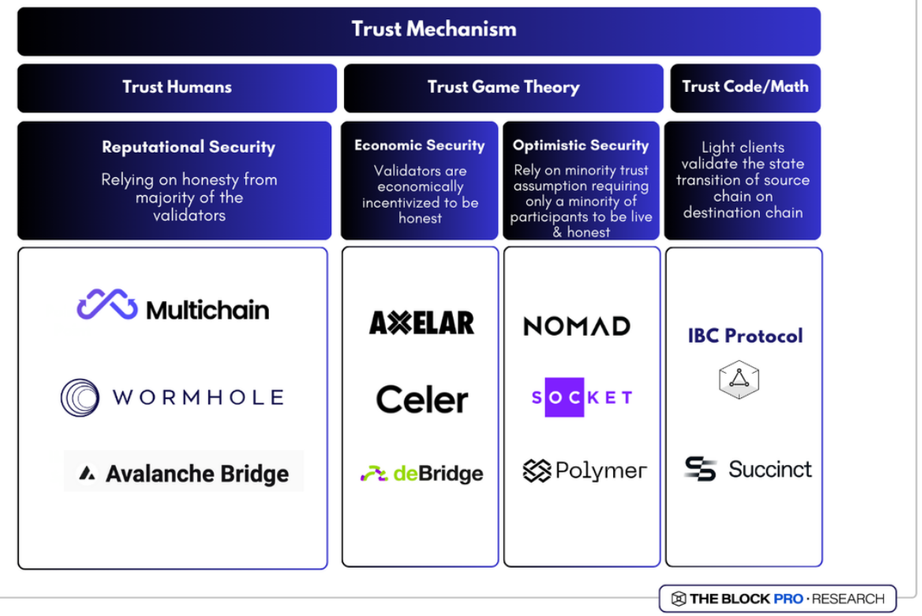
"I believe bridges should be immutable, permanent, and shouldn't be subject to any kind of third-party. Otherwise, this is not a trustless bridge. You're trusting that the people who run the code will not swap it out for some malicious code that steals funds. And nearly all bridges are guilty of this today."
- Seun Lanlege, Polytope Labs

6.1.2 RISKS AND COMPLEXITIES FOR THE VALIDATOR SET

Some bridges use a Proof of Authority (PoA) model, like Wormhole (Figure 11), where the security is based on the reputational stakes of the validators and their role in a multi-signature (multisig) system. In a permissioned system, only pre-approved validators, typically a small group of doxxed entities or companies, can stake and participate in the verification process. This model is easier to implement since it doesn't require complex incentive designs. Validators are publicly known and are motivated to act honestly to preserve their reputation. Additionally, communication and consensus are more efficient, as the validators are few and familiar with one another. In this model, multiple entities sign and approve transactions, and once they obtain enough endorsements that meet or exceed a certain threshold, the transactions are considered valid.

This model has no cryptographic or economic guarantees as the entities managing the bridged assets have no financial risk - only reputational. It assumes most entities will prioritize their reputation over any potential financial gain from dishonesty. However, this model relies entirely on the trustworthiness of third parties and their reputation in the community to keep the process honest. Possible attack vectors include social engineering, impersonation, corruption, and/or bribery.

Figure 6.1.2.1. A table comparing the trust mechanism of different bridges



Source: BlockCrunch

One method to add security and reduce trust assumptions of externally validated systems is to require external verifiers to stake collateral as a bond before they can start verifying (the Economic Security section in Figure 11). This staking mechanism adds an extra layer of security by ensuring that if a verifier is malicious (e.g., signs an invalid block header or approves an invalid cross-chain transaction), their staked collateral can be slashed as a penalty.

In a permissionless system, any party with sufficient capital can become a validator and participate in cross-chain verification. This can increase the cost of corruption (attacking ≥ ⅔ of validators) because a permissionless system may attract a much greater number of validators than compared to most

modern permissioned systems, which typically have less than 20 validators. Beyond the absolute number of validators, permissionless systems may be more anonymous and experience greater "churn" in the validator set, making it more difficult to identify the entities/persons responsible for the validator and look to attack and/or extort them.

Economic security is critical in this system. Verifiers need to stake or bond a significant amount of monetary value, which can be forfeited if proven to have acted maliciously. For the bridge to be secure, the value staked by these verifiers must exceed the total value locked (TVL) in the bridge. If the staked value is close to the TVL, a validator could be tempted to abscond with the funds because the reward (TVL) is greater than what is at risk (the value staked). Additionally, an outside attacker could profit from specific attacks on the bridge, especially when combined with shorting the tokens on the involved chains.

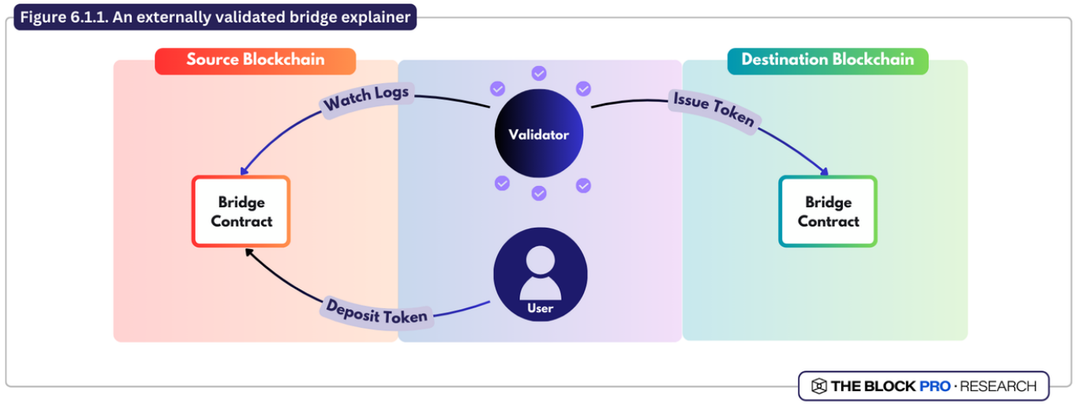
High staking requirements may be necessary to ensure adequate security. Unfortunately, this is also quite capital inefficient. Validators need to lock up significant capital upfront and leave it unused to participate in the network. PoS protocols often impose lengthy withdrawal delays to prevent validators from escaping slashing penalties after committing an offense, further tying up capital. Additionally, implementing this staking mechanism adds complexity for developers, who must design and maintain cryptoeconomic incentives to discourage dishonest behavior.

6.2 NATIVE

In response to the security issues faced by externally verified and lock-and-mint bridge designs, the cryptocurrency community is increasingly turning toward native verification for cross-chain transactions. Native verification requires each blockchain's own validators to integrate with the consensus mechanisms of other chains (Figure 12), eliminating the need for external verifiers and reducing the risk of exploits common in lock and mint models.

The most secure bridge design minimizes trust assumptions, ensuring that the destination chain inherits the security properties of the origin chain. This is achieved through on-chain verification, which can be accomplished in multiple ways. The most secure yet expensive and least scalable method is for the bridge to run a full node on every connected chain. This design has proven too onerous or cost-prohibitive for most bridge teams (aside from Wormhole). Therefore, many have turned to running light clients, where the destination chain's validators run a light client to verify the transaction's validity.

Figure 6.1.1. An externally validated bridge explainer



Source: Xangle

> "Our verifiers, what we call Guardians, run full nodes on every single chain that they're connected to. It's a very, very difficult business, and as a consequence, it has been our biggest differentiator with respect to security offering compared to every other bridge."
>
> -Maher Latif, Wormhole Foundation

However, this approach is challenging to scale (Figure 13), as requiring each validator to maintain a light client for every new blockchain connection adds greater and greater computational load for validators. Light clients enable tracking events and state changes without the need for the extensive hardware and data storage required by full nodes. By trusting the design of a light client, the protocol can download only block headers to verify specific information, such as deposits, withdrawals, or the status of message requests and executions in a messaging protocol.

Off-chain agents, called relayers, monitor events on the source chain, generate cryptographic proofs, and send them to the light client on a destination chain. This allows the light client to authenticate transactions using stored block headers that contain the Merkle root hashes needed for state verification.

Key Features:

- **Security**: The light client's initialization makes a minimal trust assumption as it starts with a specific block header from the other chain. This is what's known as a "weak assumption," as it can be verified independently by anyone. However, the relayer still relies on sending information, introducing a liveness assumption.

Figure 6.2.2. Comparing full nodes vs light client trust implementations in a bridging context

	Full Node	Light Client
Downloads Blocks	✓	✗
Validates State Transitions	✓	✗
Resource Requirements	High	Low
Security Guarantees	High	Medium
Trust Assumptions	Trustless - Independently validates blockchain	Trust Required - Relies on honest majority of full nodes

THE BLOCK PRO · RESEARCH

Source: Delphi Digital

- **Implementation**: The light client implementation varies depending on the cryptographic primitives supported by the connected chains. The light client implementation is the same for chains with the same framework and consensus algorithm, such as those using the Cosmos SDK. For chains with different architectures, such as Polkadot's Substrate and Cosmos SDK, you need a unique setup like a Tendermint light client on Substrate.

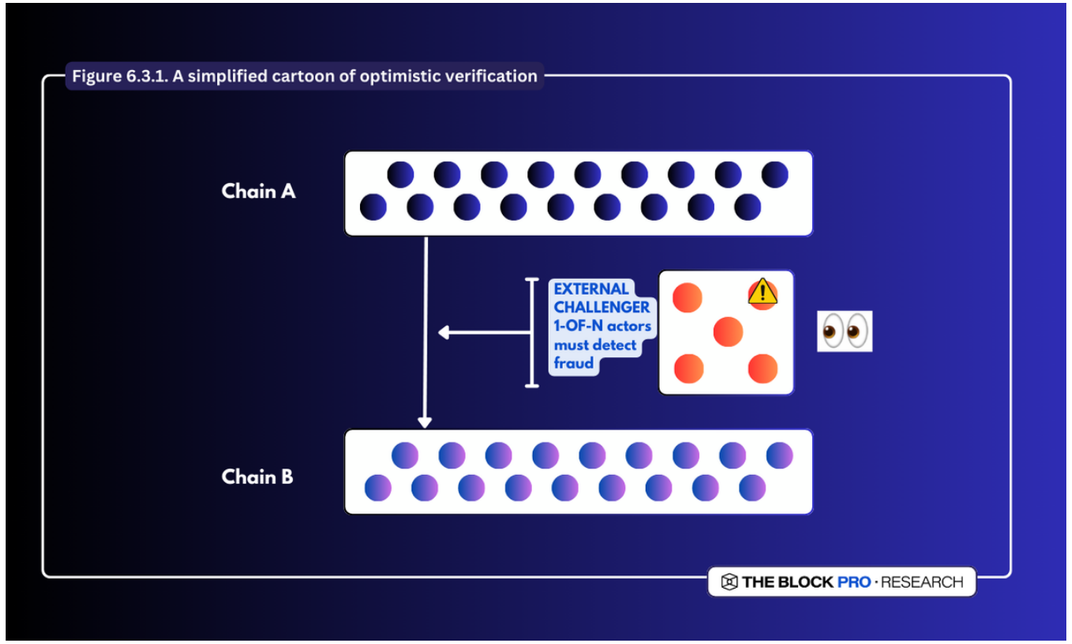
- **Challenges**: Resource intensity is a big concern as running light clients across multiple chains can be expensive (but less expensive than running full nodes), especially in environments with dynamic validator sets like Ethereum. The architectural differences between chains also limit the scalability of light clients, making inter-chain connectivity harder.

However, native verification is not a panacea. Implementation costs are high, and the actual transaction finality times can be greater than alternative methods like intent-based bridging (discussed in future sections). Despite these obstacles, the move toward native verification represents a crucial step in enhancing the security and stability of blockchain bridges.

6.3 OPTIMISTIC

Optimistically verified bridges occupy a middle ground between native and external implementations in terms of security. Similar to Optimistic rollups, these bridges assume that transactions are valid by default, allowing a challenge period during which monitors can dispute transactions if they suspect fraud or invalidity. If a challenge is successful, the transaction is reversed. The security of this system depends on at least one honest validator (1-of-n) raising the alarm within the fraud-proof window (Figure 14).

Figure 6.3.1. A simplified cartoon of optimistic verification



Source

One of the key strengths of optimistically verified bridges is their resilience during a hack. In externally verified systems, if the private keys of a majority of validators are compromised, as in the case of the Ronin bridge hack, an attacker could steal all the funds from the bridge. However, even if an attacker gains control of all validators' private keys in an optimistically verified system, they cannot be guaranteed to escape with the funds so long as a single honest watcher detects the fraud. If identified, the attacker's access to the

funds can be revoked. The downside to this approach is the extended challenge period, which results in a slower time to finality.

This approach is commonly used in optimistic roll-ups, and examples in the interoperability space include Nomad and ChainLink CCIP. Nomad allows whitelisted watchers to prove fraud, while ChainLink CCIP is developing an Anti-Fraud Network using decentralized oracle networks to monitor for malicious activities. It should be noted that in August 2022, a code bug was introduced during a routine upgrade of Nomad's bridge, allowing cross-chain messages to be auto-approved. Over the course of an hour, ~$190 million was stolen through hundreds of transactions as attackers easily replicated the exploit by copying the initial transaction, replacing the recipient address, and draining funds. This is a prime example that despite the bridge aiming for and having been designed with certain security features, a simple error in the code can corrupt the entire security design. The Nomad hack was an example of a programming error (further discussed in Section 7.1) and not in the actual security design.

Generally, if an optimistic bridge has no bugs, a common implementation involves using a middle chain with its own validators, who monitor the source chain and reach consensus on transaction validity. After consensus is achieved, these validators provide attestations on the destination chain. Axelar Network uses this approach, where validators stake tokens that can be slashed if they act maliciously.

Game-theoretic (or optimistic) approaches to cross-chain interoperability offer resource optimization, as most verification occurs off-chain, reducing on-chain resource demands. These mechanisms are also extensible, allowing the same consensus mechanism to be applied across different blockchain types, including heterogeneous blockchains. However, challenges remain. Trust assumptions in economic security mechanisms can be exploited if validators collude. Additionally, optimistic security solutions introduce complexities related to transaction finality and liveness, as users and applications must wait for the fraud window to close to ensure transaction validity.

6.4 LOCAL

Locally verified bridges enable peer-to-peer cross-chain transactions through mutual state verification between counterparties. In this model, both parties (validators on each chain) independently confirm each other's validity, executing the transaction only if both agree, minimizing the need for external validators. The core mechanism, atomic swaps (discussed previously), ensures that assets are exchanged between blockchains without intermediaries, offering high trust minimization. In this system, either both transactions succeed or both fail.

However, local verification's main limitation is that it's primarily useful for transferring tokens between two chains, not general message passing/cross-chain contract calls. Other methods would be more appropriate for complex interactions, such as creating a dApp that operates across multiple networks.

PART 7
RISKS AND CHALLENGES OF MODERN-DAY BRIDGES

No matter how complex a bridge or interoperability solution is, smart contracts are still at the heart of every design. The same complexity that makes them so convenient and useful also exposes them to significant security risks. This section delves into the vulnerabilities within smart contracts, such as parameter validation flaws and reentry attacks, and explores strategies like formal verification and multi-signature (multisig) wallets to mitigate these risks. It also discusses governance mechanisms, liquidity challenges, and cryptoeconomic considerations that impact the stability and security of cross-chain bridges. Understanding these factors is crucial for developers and users to build trust in decentralized systems and navigate the growing landscape of multi-chain blockchain networks effectively.

7.1 SMART CONTRACT ERRORS

> "Bridges tend to have two main attack vectors: operator attacks or software attacks... The biggest ones (attacks) tend to be on the software side.... due to bugs."
>
> - Sunny Aggarwal, Osmosis

Smart contracts are the backbone of any cross-chain bridge and are essential for their security and efficiency. However, smart contracts are also complex, making them a prime target for many security bugs and vulnerabilities. Insufficient validation of transaction parameters is a common vulnerability in blockchain systems, as was the case in the Multichain (formerly Anyswap) incident. In that case, the function responsible for swaps with a permit failed to validate the token parameter. This oversight allowed an attacker to exploit the system by bypassing the address of a previously deployed malicious contract. The bug allowed the attacker to steal funds from any user who had approved the bridge contract.

Several strategies must be employed to mitigate these risks. First and foremost, the integrity of smart contract execution must be defended. This involves ensuring that functions within the contract rely on predefined conditions, which must be met before execution can occur. For example, checking the balances of bridge liquidity contracts is crucial to prevent actions from taking place when insufficient funds exist.

Additionally, formal verification and thorough testing of smart contracts are needed to ensure they are correct under any conditions. This means validating the contracts' logic and testing thoroughly, especially after updates, to make sure everything works as expected.

When designing a smart contract, developers have to decide if the contract should be upgradeable or non-upgradeable. Non-upgradeable immutable contracts prevent unauthorized changes and limit the surface for future attacks. The downside is that this immutability means any discovered vulnerabilities cannot be patched. Upgradeable contracts offer flexibility to update and fix but require careful planning to prepare for failures. One way to do this is to use proxy patterns, which separate the contract's state from its logic so you can update the logic without touching the underlying data.

Finally, addressing specific technical vulnerabilities, such as reentry attacks, is essential for ensuring the overall security of smart contracts. Preventing re-entry attacks involves using techniques like sequence numbers, timestamps, or cryptographic proofs to validate contract calls and prevent the double execution of functions.

7.2 CORRUPTING MULTISIG KEYS/SOCIAL HACKING

One of the most critical vulnerabilities in blockchain security is improper key management, particularly when these keys protect essential functions like minting or unlocking assets. If operators fail to secure their keys, attackers can access them, enabling unauthorized minting of tokens or withdrawal of all locked assets.

To mitigate this risk, many systems employ multi-signature (multisig) protocols, which require multiple keys (let's assume N keys) to authorize a transaction. In a typical setup, a subset of M keys out of N is needed to create a valid signature, ensuring that no single key can unilaterally unlock funds. This structure, known as M-of-N multisig, enhances security by distributing control across multiple parties.

While multisig wallets add an extra layer of security by requiring multiple individuals to authorize transactions, they are not without risks. These wallets, often used to enable upgrades or pause bridge contracts, are crucial for safeguarding assets. However, they also introduce additional vectors for attack due to their complexity and reliance on human operators. The security of a multisig wallet is only as strong as the practices of its signers, whose private keys must be securely managed to prevent unauthorized access. The human factor introduces a significant risk, as signers can be targeted through social engineering tactics like phishing or malware attacks, making them attractive targets for malicious actors.

The effectiveness of multi-signature (multisig) wallets can be compromised when they are improperly managed, as shown in the case of the Ronin bridge used by Axie Infinity. Originally, the Ronin bridge operated with a 5-of-9 multisig, requiring five out of nine key holders to authorize a transaction. However, four of these keys were controlled by Sky Mavis, the developer of Axie Infinity, reducing the security to a 2-of-6 multisig. The situation worsened when the Axie DAO, which controlled another key, granted access to Sky Mavis, effectively turning it into a 1-of-5 multisig. This meant that a single security breach could (and did) compromise the entire bridge. While multisigs offer more security than relying on a single individual, they still require careful management and trust in the individuals involved to prevent potential exploits.

7.3 DIFFERENT FINALITY

Finality in decentralized systems, especially blockchain networks, is key to making sure once a block is accepted, it is permanent and immutable. This concept is vital for maintaining a consistent view among network participants and preventing block reversion, which could undermine trust in the system. However, different blockchain protocols handle finality in different ways, often depending on their structure and security.

In the context of blockchain bridges, the security and reliability of transactions depend on ensuring blocks are final before any cross-chain operations are executed. This nuanced view of finality allows for more flexible and secure bridge designs for different use cases. However, the system is still vulnerable to attacks like a 51% attack on the source chain, which can compromise the bridge and the assets.

7.4 GOVERNANCE ATTACKS

Governance and access control are other vital aspects to consider in the security of smart contracts. In cases where control is decentralized and handed over to token holders, securing governance mechanisms becomes paramount to prevent large holders from manipulating the system for their benefit. Implementing processes like optimistic approval, time-locks, and TWAP voting can mitigate the risks associated with decentralized governance, ensuring that decisions are made in the best interest of the community as a whole. Furthermore, access control must be carefully managed to define who can access specific contract functions. Critical functions, such as minting tokens or upgrading contracts, should not be concentrated in the hands of a single entity, as this would create a single point of failure and increase the risk of exploitation.

7.5 LIQUIDITY CHALLENGES

Current bridges often facilitate cross-ecosystem value transfers (for example, from Ethereum to Solana) by taking control of users' funds and redirecting them to the destination chain. This process is somewhat analogous to how banks manage and reroute money globally. Just as banks acquire assets and issue liabilities, many popular bridges of today must balance the deposits they control and the liabilities they incur. The solvency of these "blockchain banks" depends on their ability to remain fully capitalized while continuously processing deposits and redemptions. However, even if centralized bridges were perfectly secure, most would still be highly capital-intensive due to the need for liquidity on every integrated chain. These operational costs are inevitably passed down to users through substantial fees, reaching up to 1-2% (in extreme cases) of the transaction value. This liquidity challenge is only exacerbated as the number of independent blockchains, including Layer 1s, Layer 2s, and appchains, continues to grow.

7.6 CRYPTOECONOMIC SECURITY

While bridge-building in blockchain networks is often perceived as a technical challenge, it also presents significant economic implications. Integrating an asset into a multi-chain environment typically disrupts its original tokenomics, which may not have been designed for such exogenous factors. When a project with a token on a single blockchain expands to another chain, it must choose between inflating the total token supply or implementing some form of asset mapping.

For instance, if Uniswap's UNI token is initially issued on Ethereum, launching UNI on Polygon would require the Uniswap community to either mint additional UNI tokens on Polygon or lock an equivalent amount of UNI on Ethereum to preserve the fixed supply. Most projects prefer the latter approach to avoid inflation, but the trade-off is added complexity and fragmented liquidity.

PART 8

INTER-CHAIN COMMUNICATION IN LEADING MULTI-CHAIN PROTOCOLS

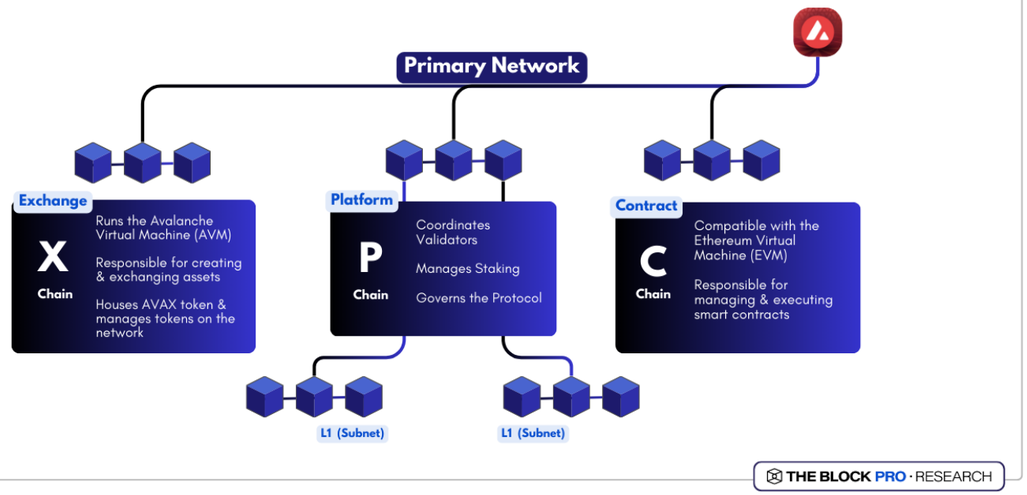
Inter-chain communication and interoperability are distinctly different from cross-chain interoperability solutions and bridges. Unlike traditional cross-blockchain communication protocols, such as bridges and interoperability protocols like LayerZero, which often rely on third-party validators or intermediaries to facilitate asset transfers between chains, inter-chain communication emphasizes direct communication channels between chains within the same protocol that preserves security and decentralization. Examples include Avalanche's Interchain Messaging (ICM), Cosmos' IBC, and Polkadot's Cross-Consensus Message Format (XCM). These protocols enable chains to share data and assets natively, fostering a more integrated and cohesive blockchain network while reducing the risks associated with cross-chain attacks and trust dependencies.

8.1 AVALANCHE

The Avalanche blockchain aims to offer a robust and scalable framework for creating custom L1 blockchain networks, formerly known as subnets, that operate in parallel with its main network. The Avalanche network operates with three main chains—Platform (P)-Chain, Exchange (X)-Chain, and Contract (C)-Chain—each serving specific functions (Figure 15). The P-Chain coordinates validators, manages staking, and governs the protocol. The X-Chain runs the Avalanche Virtual Machine (AVM) and is primarily responsible for creating and exchanging assets. The X-Chain houses the native AVAX token and manages tokens on the network. The C-Chain is compatible with the Ethereum Virtual Machine (EVM) and is responsible for managing and executing smart contracts.

Avalanche uses a Delegated Proof-of-Stake (DPoS) consensus mechanism, which, due to its efficiency, enables faster transaction processing compared to Proof-of-Work (PoW) and traditional Proof-of-Stake (PoS) systems. The AVAX token is integral to all chains and L1s within the network, ensuring unified functionality across the ecosystem.

Figure 8.1.1 An overview of Avalanche's Primary network



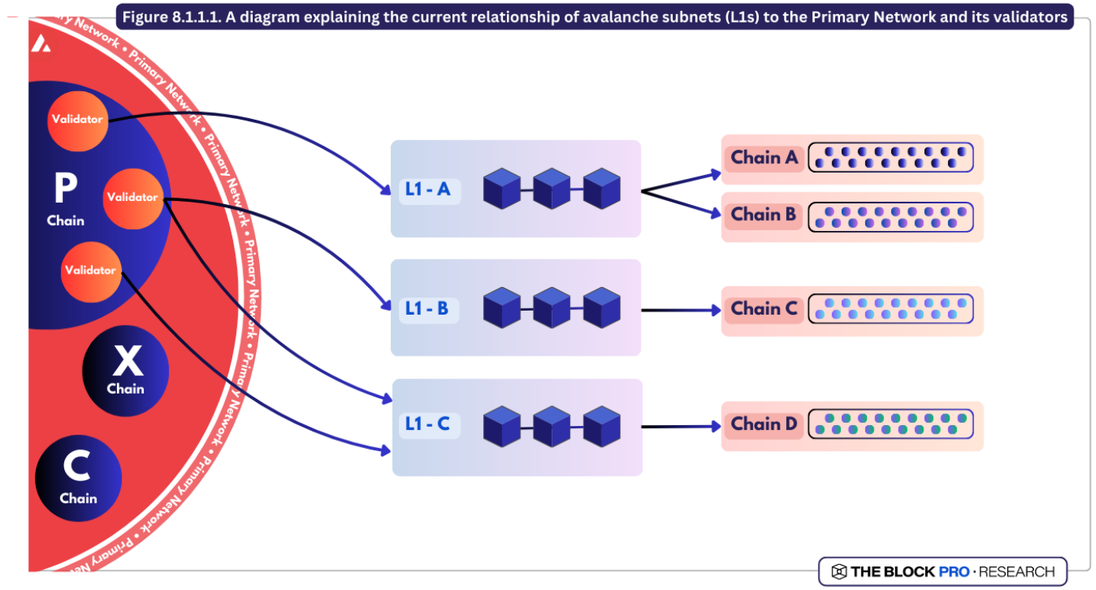
Source

8.1.1 L1S (FORMERLY SUBNETS)

Avalanche's platform extends beyond its three core blockchains and the Primary Network by allowing the creation of application-specific blockchains, known as "L1s." Unlike generic blockchains, L1s are highly customizable, fully interoperable with one another, and can operate as either permissioned or permissionless networks.

An L1 in Avalanche is a dynamic group of validators that reach consensus on a specific set of blockchains (Figure 16). A single L1 validates each blockchain, although an L1 can validate multiple blockchains. The P-Chain in Avalanche is a special L1 responsible for validating all blockchains within the Avalanche ecosystem, including the X- and C-Chains.

Figure 8.1.1.1. A diagram explaining the current relationship of avalanche subnets (L1s) to the Primary Network and its validators



Source

These L1 blockchains are connected via the P-Chain, enabling them to scale horizontally. Instead of launching smart contracts on existing networks like Ethereum, developers can launch sovereign chains with more flexibility and customizations than would otherwise be possible. The platform can also support up to 4,500 TPS per L1 and, theoretically, even higher TPS as more validators join the network.

Avalanche's L1s are similar to Ethereum's rollups and Cosmos's zones/hubs but with key differences. In Cosmos, each zone/hub has its own validators, whereas in Avalanche, one L1 can validate multiple blockchains with one validator set. L1s in Avalanche can have their own virtual machines, programming languages, tokens, fee structures, validator sets, slashing rules, and more.

The current hardware requirements to validate an Avalanche L1 are quite onerous for some, especially considering the operational costs and technical requirements. Validators need to validate the entire Primary Network of Avalanche, which includes the Contract Chain (C-Chain), Platform Chain (P-Chain), and Exchange Chain (X-Chain). This requires a lot of resources: at least 8 AWS vCPUs, 16 GB of RAM, 1 TB of storage, and a minimum stake of 2,000 AVAX tokens. While this was manageable initially, the rise of

AVAX's value has made it a financial burden, peaking at over $250,000 during previous market highs and now around $55,000-$60,000.

To address these challenges, ACP-77 was proposed to overhaul Avalanche L1 validation and make it more accessible and autonomous. The proposal suggests that validators would no longer need to validate the entire Primary Network. Instead, they would only need to sync with the P-Chain, which tracks their specific L1 validator set and enables cross-L1 communication. This would greatly reduce the operational and staking costs and make it easier for new validators to join.

In addition, ACP-77 would allow Avalanche L1s to set their own validation rules and staking requirements, giving back sovereignty to the L1s and enabling horizontal scaling - where multiple independent blockchains can run concurrently. This flexibility is particularly advantageous for regulated entities, which could opt to validate only their permissioned networks, avoiding the complexities and risks of validating the entire permissionless Primary Network.

Additionally, the proposal introduces a dynamic fee mechanism to replace the fixed transaction fees on the P-Chain. This dynamic fee would be based on the number of registered validators and network utilization, adjusting as needed to maintain economic sustainability. Validators would need to maintain balances on the P-Chain to cover these ongoing fees, ensuring continuous contributions to the network without the high upfront costs.

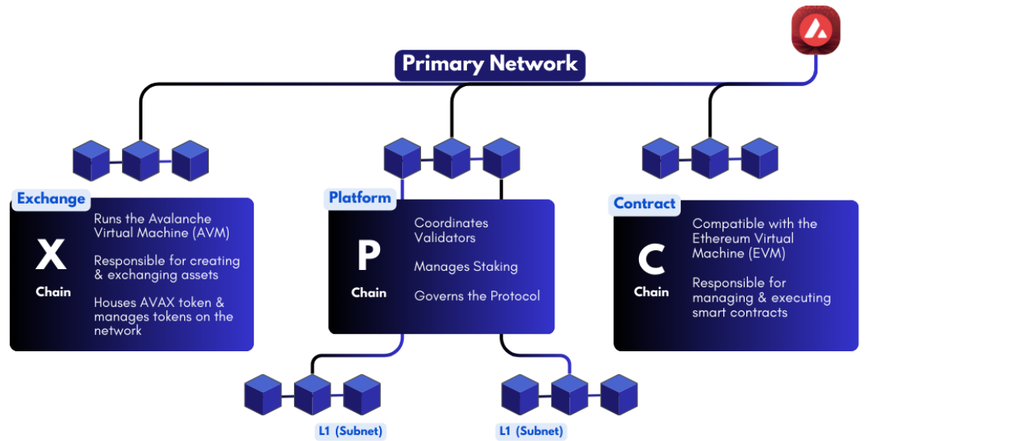
8.1.2 INTER-CHAIN MESSAGING (FORMERLY AVALANCHE WARP MESSAGING)

The activation of the Durango upgrade on the Avalanche mainnet in March 2024 marked a significant leap forward in enhancing interoperability within the Avalanche ecosystem. This upgrade introduced the Avalanche Warp Messaging (AWM) through the Teleporter protocol, enabling seamless cross-L1 communication for the first time on Avalanche. The Teleporter, compatible with the Ethereum Virtual Machine (EVM), facilitates native cross-L1 smart contract interactions, creating a unified network where L1s can communicate effortlessly. By enabling cross-L1 messaging and transactions, the Teleporter creates a unified liquidity pool across the network, unlike the liquidity fragmentation seen with Ethereum rollups. L1s enjoy this unified liquidity while also maintaining the autonomy and customization of individual blockchains.

The Teleporter's key advantage lies in its reliance solely on the validators within Avalanche's Primary Network and the participating Layer 1s (L1s) for trust, eliminating the need for third-party intermediaries. This setup ensures a secure and efficient mechanism for direct L1-to-L1 communication (as shown in Figure 17), with the P-Chain serving as a central validator set registry, simplifying operations while enhancing security.

When one Avalanche L1 processes a message from another, it verifies the message using the origin L1's BLS public keys and stake information. This allows the receiving L1 to confirm the message's authenticity via signature verification. The validation threshold for these messages can be customized according to the requirements of each L1. For instance, L1 A might accept messages from L1 B if they are signed by validators representing at least 67% of the stake, while it might require an 85% stake threshold for messages from L1 C.

Figure 8.1.1 An overview of Avalanche's Primary network



Source: Xangle

Since the P-Chain maintains an up-to-date record of all validators' public keys and stake weights, this information is readily available to any virtual machine operated by the validators. As a result, Avalanche L1s do not need to communicate directly regarding changes in validator sets. Instead, they rely on the P-Chain's current data, ensuring that AWM introduces no additional trust assumptions beyond the integrity of the origin L1's validators.

However, the Teleporter has its limitations. It currently operates only between EVM-compatible chains within the Avalanche ecosystem, limiting direct interoperability with Ethereum and other major blockchains. Moreover, as a newer technology, it may introduce unforeseen security risks and could face performance challenges under high transaction volumes. Despite these potential drawbacks, the Teleporter represents a significant advancement in cross-chain interoperability within Avalanche.

8.2 COSMOS

The Cosmos Network was originally designed with interoperability in mind. Through its unique “hub-and-zone” architecture and the implementation of the Inter-Blockchain Communication (IBC) protocol, Cosmos connects sovereign blockchains, enabling them to communicate and interact seamlessly. At the core of Cosmos lies the Cosmos Hub, a blockchain that operates using a Proof-of-Stake (PoS) consensus mechanism. This hub serves as the primary chain within the Cosmos ecosystem, where the native ATOM token resides. Stakers of the ATOM token play a critical role in securing the network by validating and routing transactions across various blockchains, referred to as "zones" within this ecosystem. This model positions the Cosmos Hub as a coordinating chain, akin to a beacon chain, that connects and integrates diverse zones, each with its own unique set of features and capabilities.

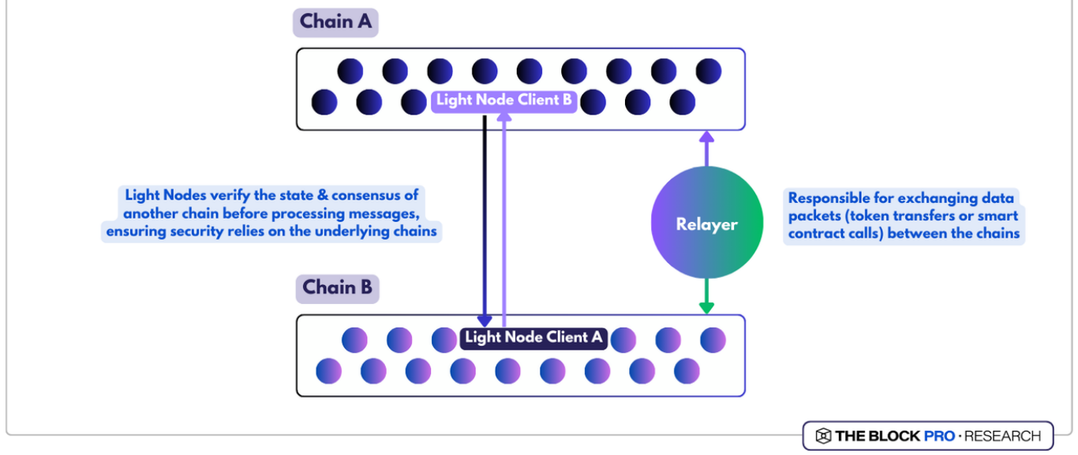
Cosmos also offers app-specific blockchains, often called "appchains," designed to meet the specific needs of Web3 protocols. These appchains can support a variety of features, including data availability, NFT integration, interchain operability, and wallet support. One of the standout components enabling this functionality is the Inter-Blockchain Communication (IBC) protocol.

8.2.1 IBC

The IBC protocol is an asynchronous messaging system that allows for secure and trustless communication between the blockchains in the Cosmos ecosystem. It enables the exchange of data, messages, and tokens using light clients to make inter-Cosmos communication more efficient and trustless. These light clients verify the state and consensus of another chain before processing messages, ensuring security relies on the underlying chains. However, these benefits are limited to Cosmos chains that use Tendermint/Comet Consensus.

For most blockchains within the IBC network, each participating blockchain must deploy a light client that cryptographically verifies the consensus state of the other chain (Figure 18). This verification is key to each chain knowing the state of the other and to the integrity and security of the communication. A relayer, a part of the IBC protocol, is responsible for exchanging data packets (token transfers or smart contract calls) between the chains.

Figure 8.2.1.1. An illustration of the relayers and light client roles in the IBC



Source

The relayer's primary function is to submit messages to the IBC module on each chain, ensuring that light clients remain in sync with the current state of the connected blockchain. In this system, relayers, not users, pay for the cross-chain communication. One drawback to this approach is that it can reduce liveness temporarily if relayers fail to operate. However, more generally, the trust assumptions within this system are rooted in the validator sets of the participating blockchains, with the overall security of the IBC

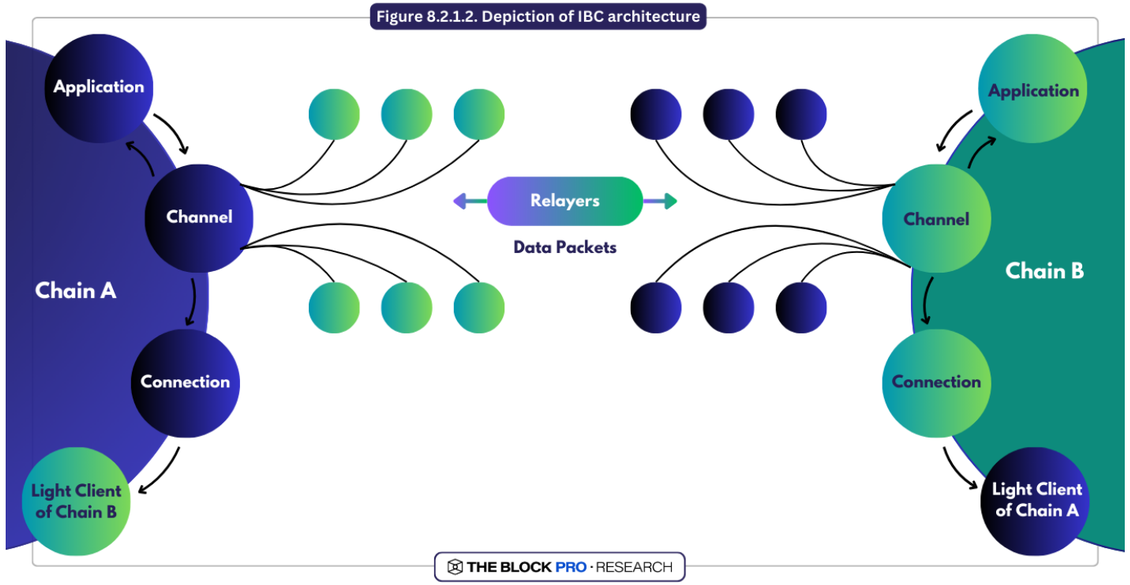
process being dependent on the cryptoeconomic security provided by the total stake of each blockchain involved.

> "IBC was built to be as neutral as possible. The way IBC is designed is that it allows the team/developers to utilize different authentication mechanisms. We currently try to use light clients as much as possible, but if better authentication methods become available, like zero-knowledge proofs, you can plug that into the IBC as the new authentication mechanism."
>
> - Sunny Aggarwal, Osmosis

To establish an IBC connection between two distinct chains, a series of initial transactions must be executed on each blockchain. These transactions contain critical information and state proofs from the other chain, allowing each blockchain to authenticate the other's light clients in preparation for data exchange.

Before the chains can fully interoperate, they must open channels between them (Figure 19), which involves exchanging data packets. Once the channels are open, the chains can communicate. Each IBC connection can be configured with specific settings, such as timeout or ordering guarantees, depending on the needs of the participating chains.

Figure 8.2.1.2. Depiction of IBC architecture



Source: Binance Research

The process of transferring tokens via IBC follows a precise order:

1. Tokens on the originating chain (Chain A) are locked in escrow, and an outgoing data packet is produced detailing the sender, receiver, and amount.
2. The relayer transfers the data packet to the destination chain (Chain B), where it is verified.
3. Upon successful verification, new tokens of an equivalent amount are minted and transferred to the receiver on Chain B.
4. Chain B's IBC module generates an acknowledgment packet confirming the receipt of tokens.
5. The relayer transfers this acknowledgment packet back to Chain A, completing the transfer process.

Cosmos's approach to blockchain interoperability offers several key advantages, particularly when compared to traditional models like Ethereum's smart contract-based architecture. By enabling projects to launch as sovereign appchains rather than as a set of smart contracts on an existing platform, Cosmos offers a level of customization and control often lacking in other ecosystems.

Key benefits include:

1. **Customizable Validator Rules**: Appchains on Cosmos can define their own validator rules and requirements, allowing them to tailor their security and governance structures to meet their specific needs, whether that involves public versus private chains or privacy versus transparency.
2. **Performance Isolation**: By operating as independent blockchains, appchains are insulated from the performance issues of other projects, ensuring that one decentralized application (dApp) does not disrupt the entire chain.
3. **Predictable and Customizable Fees**: Appchains can establish their own fee structures, providing greater predictability and control over transaction costs, which can be a significant advantage in managing network economics.

While the IBC protocol provides a secure and efficient way for blockchain interoperability within the Cosmos ecosystem, it's not without its challenges. Some of the inherent limitations of IBC include maintaining light clients across many chains and connecting chains with different consensus algorithms.

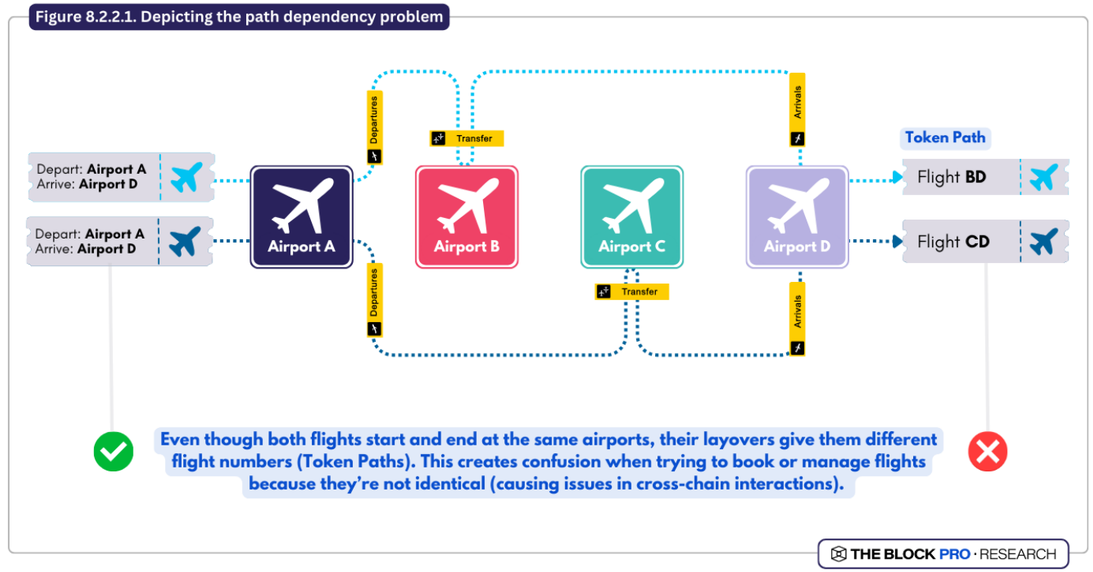
One of the primary challenges with the IBC model lies in its scalability. Creating and maintaining light clients across potentially hundreds of chains is cumbersome. Light clients are “lighter” than full nodes but still require considerable resources to operate at scale. This complexity increases as the number of interconnected chains in the Cosmos ecosystem grows. IBC assumes the connected chains use the same consensus algorithm, Tendermint, which has deterministic finality. Deterministic finality makes bridging these chains easier as the state of the blockchain is final as soon as a block is completed. However, connecting to chains with probabilistic finality, like Ethereum, is more troublesome. Probabilistic finality, where the state of the blockchain is not instant final, creates a double-spend attack risk that doesn’t exist in traditional IBC-connected zones.

To address the challenges associated with connecting to probabilistic chains, Cosmos employs a concept known as "Peg Zones." A Peg Zone is a specialized blockchain designed to monitor the state of a probabilistic chain and establish a point of finality. This zone only interacts with other IBC zones once enough time has elapsed to minimize the risk of a chain reorganization, which could otherwise compromise the security of cross-chain transactions. Peg Zones serve as an intermediary between deterministic and probabilistic chains, ensuring that the Cosmos network can maintain interoperability without compromising on security.

8.2.2 IBC PATH DEPENDENCY

Another significant challenge faced by the IBC protocol is the issue of path dependency (Figure 20), which can reduce the fungibility of tokens and create friction in cross-chain transactions. Prior to improvements implemented in 2023 (discussed below), a token's value could vary depending on its route through different chains. For instance, a token routed from Moonbeam to Osmosis to the Cosmos Hub is different from one routed from Moonbeam to Axelar to the Hub despite originating and ending in the same locations. This is because each token's path is encoded in its denomination, which uses SHA256 to produce a fixed-length output. As a result, the ICS-20 module in IBC must maintain a mapping of all token denominations since it cannot compute the original input to trace the token's path. While this design enhances security by ensuring that tokens with different paths reflect different security guarantees, it complicates user experience and cross-chain interactions, particularly for decentralized exchanges (DEXes) beyond Osmosis, where tokens may have traveled through various chains.

Figure 8.2.2.1. Depicting the path dependency problem



Source

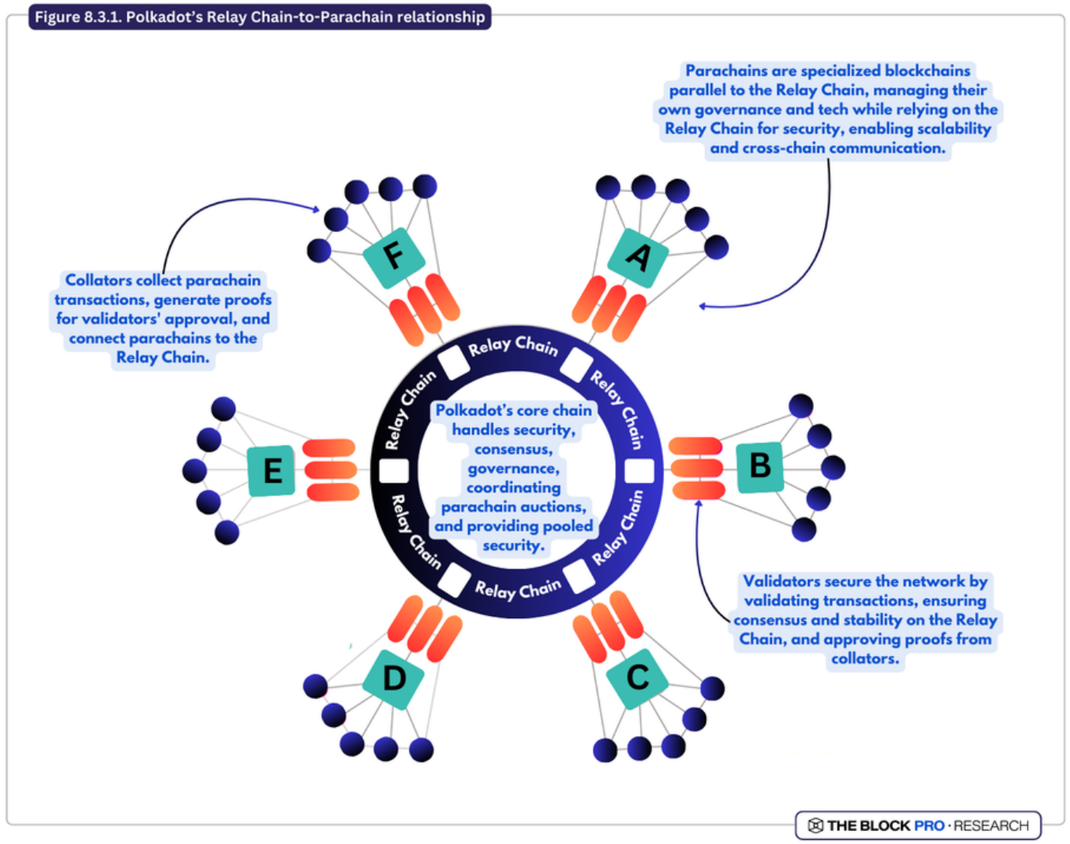
Cosmos is working on several solutions to address the path dependency issue. One of them, "path unwinding," is a mechanism that allows tokens to go back to their original chain before reaching their final destination. This adds a layer of latency but reduces the number of hops a token has to go through, making it more straightforward and fungible. Path unwinding is especially useful when there are multiple transitions across multiple chains, as it bridges tokens directly to their native chain before going to the final destination.

At the same time, Cosmos is also introducing the Packet Forward Middleware, a solution specifically designed for IBC. This middleware allows blockchains to route IBC packets through the source chain first to make the token denomination uniform and fungible across transactions. By centralizing the routing through the source chain, the Packet Forward Middleware simplifies multi-hop IBC transactions and turns complex multi-chain interactions into a single transaction. But both path unwinding and Packet Forward Middleware put additional load on IBC relayers, so relayer incentivization is key.

8.3 POLKADOT

The Polkadot network is similar to Cosmos in that it is designed to enable cross-chain communication and interoperability by connecting multiple blockchains into a unified system. However, its structure consists of a main Relay chain and various Parachains (Figure 21), aiming to achieve horizontal scalability through an asynchronous heterogeneous network model. Polkadot's modular architecture is centered around pooled security, community-driven governance, and seamless interoperability among its parachains.

Figure 8.3.1. Polkadot's Relay Chain-to-Parachain relationship



Source

Polkadot enables its parachains to be app-specific (appchains), run on various virtual machines, and interoperate when necessary. The current Polkadot 1.0 framework requires parachains to secure slots through competitive auctions, demanding significant DOT collateral. This, in theory and in practice, can prove to be a barrier for smaller projects looking to launch on Polkadot. Recently announced but yet to be released, Polkadot 2.0 addresses these issues with Agile Coretime, which dynamically allocates computational resources based on demand, enhancing efficiency, accessibility, and scalability without compromising security or decentralization. The transition will begin with trusted or permissioned collators and gradually expand to include untrusted or permissionless collators, culminating in the full integration of Elastic Scaling across the network.

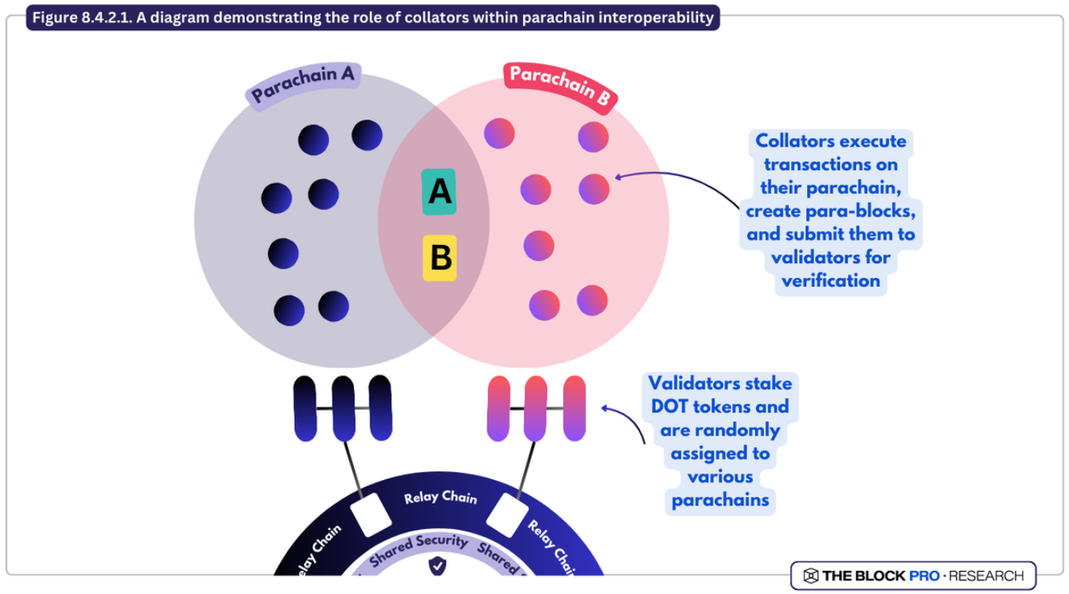
8.3.1 PARACHAINS

Parachains are specialized blockchains that run parallel to Polkadot's Relay Chain, which handles the network's security, consensus, and interoperability. While the Relay Chain coordinates the overall system and supports essential transactions like governance and parachain auctions, it does not support smart contracts. Parachains make a singular trust assumption by relying on the consensus of the Relay Chain. This shared reliance is central to the functioning of these networks, allowing the parachains to focus on their own governance, economics, and technological decisions. They offer scalability by running transactions independently of the main chain and support interoperability by enabling communication and data transfer between parachains.

8.3.2 POLKADOT SHARED SECURITY

Polkadot originally distinguished itself within the blockchain ecosystem through its innovative shared security model. Unlike networks such as Cosmos or Avalanche, where each independent chain was originally required to bootstrap its own validator set, Polkadot enables disparate parachains to operate securely without independently securing their networks.

At the core of Polkadot's security architecture lies the Relay Chain, where validators stake DOT tokens and are randomly assigned to various parachains. These parachains function similarly to child chains, each managed by a collator. Collators execute transactions on their parachain, create para-blocks, and submit them to validators for verification (Figure 22).



Source

Verifying a block's Proof of Validity (PoV) is computationally expensive. Therefore, para-validators are incentivized for their work. Once a para-validator group approves a block, a cryptographic commitment to the block is sent to the Relay Chain, where it is included and finalized upon receiving majority approval from the Relay Chain's validator set.

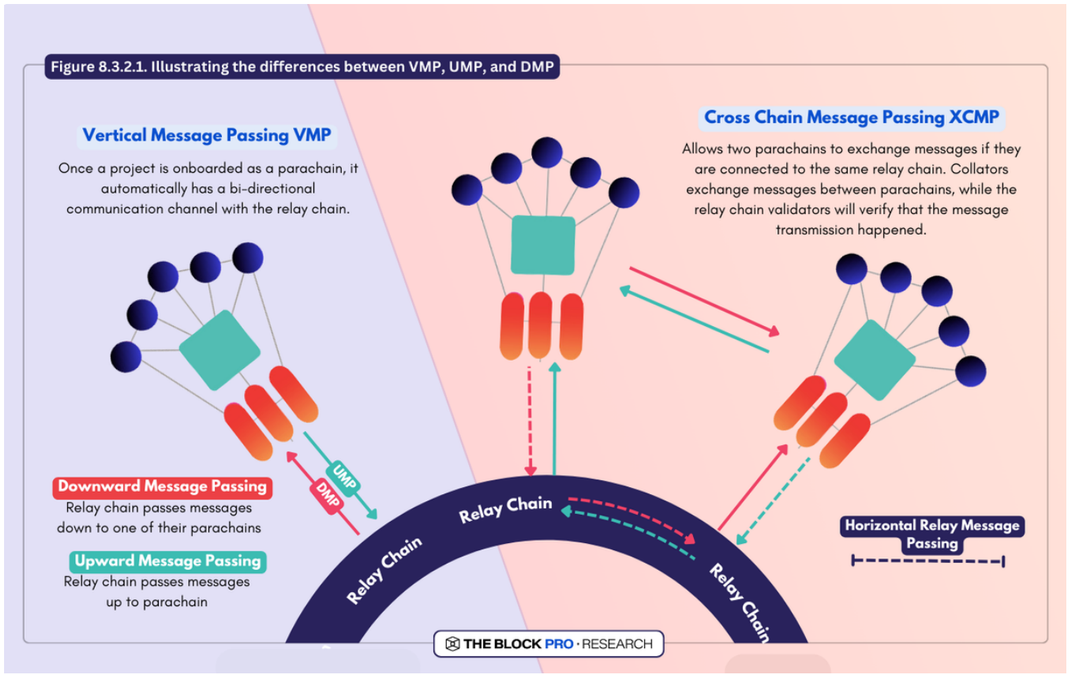
Since each parachain has a relatively small number of validators, the risk of corruption is mitigated by a secondary check done by another set of randomly chosen nodes. This redundancy makes the network more secure. If a block is identified as invalid or partially unavailable, a dispute can be initiated on the Relay Chain, where all validators re-execute the disputed block. The dispute is resolved with a 2/3 supermajority vote, and validators found to be at fault are slashed on-chain. This ensures the integrity of the entire Polkadot ecosystem, even if individual parachains are not secure.

8.3.3 XCM

Polkadot has developed a versatile framework for cross-parachain communication called the Cross-Consensus Message Format (XCM). This framework enables interoperability across various blockchain

entities, including parachains, smart contracts, bridges, and Substrate pallets. XCM functions as a standardized messaging language that allows different blockchain systems to interact seamlessly. While XCM is often associated with parachains, its utility extends to any consensus system that reaches finality to determine the correct state, whether it's a Polkadot parachain, an EVM smart contract, or another bridged system.

Due to parachains' reliance on the Relay Chain for security, the direction of message passing within Polkadot matters. There is a technical distinction between messages from the Relay Chain to parachains (Downward Message Passing or DMP) and messages from parachains to the Relay Chain (Upwards Message Passing or UMP) (Figure 23).



Source

XCM is designed to operate seamlessly with Vertical Message Passing (VMP), enabling efficient message exchanges between the Relay Chain and parachains. Moreover, it incorporates Cross-Chain Message Passing (XCMP), allowing parachains within the same Relay Chain to communicate directly.

The life cycle of a cross-chain message within the XCM format involves several key steps that ensure secure and reliable communication across different parachains. The steps are listed below.

1. Message Creation: The user or application on a parachain initiates the cross-chain operation by generating an XCM message.
2. Submission to Local Parachain: The XCM message is processed and prepared for transmission according to the local parachain's rules.
3. Relay Chain Involvement: The message is forwarded to the Polkadot Relay chain, which interconnects all parachains and facilitates their communication.
4. Message Routing: The Relay chain routes the message to the target parachain, ensuring it reaches the correct destination.
5. Receipt by Destination Parachain: The target parachain interprets the message according to its specific logic.
6. Execution of Requested Operation: The destination parachain performs the requested operation, such as token transfers or smart contract executions.
7. Feedback Loop: If needed, a response or confirmation is sent back to the original parachain.
8. Finalization: Changes from the cross-chain operation are confirmed and reflected on both the originating and destination parachains.

Note that XCM does not directly send messages between systems. Instead, it defines the format of how those messages are sent. It's important to understand that XCM messages are not onchain transactions. They only describe the intended state changes on the target network, but the messages themselves do not execute those changes.

This concept is linked to asynchronous composability, which allows XCM messages to operate independently of time-bound processes like on-chain scheduling. This means that messages can be executed in the intended order without being constrained by specific timing mechanisms on the blockchain.

At the core of the XCM format is the Cross-Consensus Virtual Machine (XCVM), a highly specialized, non-Turing-complete virtual machine. In XCM, a "message" is a program running on the XCVM that can contain one or more XCM instructions. The programs execute sequentially until they complete or error out, at which point they halt.

One of the most common use cases for inter-chain messaging is to transfer assets between chains. However, this is not its only use case. XCM enables complex, multi-hop, multi-network communication across blockchain ecosystems. One of its key features is programmability, which allows developers to set specific expectations for messages. This enables more sophisticated and diverse use cases for user and more flexibility for developers.

8.3.4 XCMP

The XCM protocol is key to secure and native interoperability between parachains. One of its main features is XCMP messages, which are signed by the same validators that secure the involved chains. These validators are full nodes plus nodes specific to the parachains that manage the state transitions. This shared validation means arbitrary data can be sent between parachains safely and with unified finality, where the network agrees upon the final state.

In XCMP, the Relay Chain plays a crucial role. Validators from the Relay Chain and the individual parachains need to authenticate messages between parachains. The process is complete when the Relay Chain includes a block that verifies and embeds the message into the network. This means the Relay Chain needs to be constantly monitored for state changes to achieve consensus on each message. The Relay Chain essentially serves as a trusted third party, and the entire system's security is fundamentally linked to the integrity of the Relay Chain's validator set. Consequently, parachain security is derived from a shared security model where XCMP's cryptoeconomic security is backed by the total stake of the Polkadot network. This model relies on the network's total stake and security mechanisms to make a collective trust assumption.

Key properties of XCMP include:

- **Trustless design**: No additional trust layers are required; the same validators that secure a parachain also ensure correct message passage between parachains.
- **Correct message ordering**: Input/Output validation mechanisms ensure messages are processed in the proper sequence.
- **High availability**: Messages are reliably accessible and recoverable using distributed erasure-coded pieces.
- **Consistency**: Guarantees that received messages are exactly as sent, even during chain reorganizations.
- **Efficiency**: Minimizes bandwidth overhead, ensuring quick and responsive cross-chain communication.

8.4 COMPARING COSMOS, AVALANCHE, AND POLKADOT

Avalanche L1s excel in scenarios where flexibility and performance are key. They enable projects to operate as separate blockchains while choosing their own consensus and governance models (Figure 24). This flexibility, combined with greater security customization due to validators' ability to connect to whatever mix of chains within the Avalanche network, makes L1s highly adaptable.

Cosmos Layer 1s are best suited for projects that prioritize cross-chain interoperability. The Cosmos ecosystem offers seamless communication between blockchains through its Inter-Blockchain Communication (IBC) protocol, making it ideal for dApps requiring native interoperability. However, launching an appchain on Cosmos and setting up IBC connections with other chains requires an increased level of technical expertise and development effort among the options, which could be a barrier for some projects.

Polkadot parachains are superior when easy integration and streamlined development are priorities. They benefit from the security and interoperability provided by the Relay Chain and are ideal for projects that can secure the necessary capital to win a parachain slot through auctions. However, the need for significant capital in auctions and limited cross-chain compatibility are notable disadvantages that aim to be addressed in upcoming upgrades.

Figure 8.4.2. A Comparison of Polkadot, Avalanche, and Cosmos Properties

	Parachain	L1	Layer 1
	Polkadot	Avalanche	Cosmos
Purpose	Shared security & interoperability across blockchains.	Customizable blockchains with independent validators & tokens.	Standalone blockchain network focusing on interoperability & customization.
Use Case	Ideal for projects needing interoperability within Polkadot with shared security.	Perfect for custom blockchain solutions with specific governance & tokenomics.	Best for autonomous, customizable projects with inter-blockchain communication.
Scalability	Scalable via parallel processing on parachains.	Scalable with multiple L1s processing transactions independently.	Scalable through application-specific blockchains & IBC protocol.
Consensus	Nominated Proof-of-Stake (NPoS).	Avalanche's consensus mechanism.	Varies, often Tendermint (BFT).
Interoperability	High within the Polkadot ecosystem via the Relay Chain.	High within the Avalanche ecosystem.	High across chains via IBC for secure interchain communication.
Customization	High, with tailored blockchain designs leveraging shared security.	Very high, enabling custom rules, tokenomics, & validator sets.	Independent security per chain with staked validators and/or shared/replicated security in Cosmos 2.0
Security Model	Shared security via the Relay Chain.	L1s manage their own security, with optional use of primary network validators (possibly changing with ACP-77)	Very high, with chains customizable for specific use cases.

THE BLOCK PRO · RESEARCH

8.4.1 MOVEMENT TOWARDS SHARED SECURITY

Shared security refers to a model where a protocol derives its security from an external source, typically by leveraging the resources of another, more established network. This is in contrast to the traditional model, where each blockchain secures itself independently. In shared security models, the capital or computational power of participants in one protocol is used to create economic security for another, usually newer/smaller protocol.

In traditional public blockchains like Bitcoin and Ethereum, consensus algorithms are combined with Sybil-resistance mechanisms—such as Proof of Work (PoW) or Proof of Stake (PoS)—to ensure network liveness and raise the cost of various attacks, including Sybil attacks, long-range attacks, and eclipse attacks.

These mechanisms rely heavily on the network's internal resources to maintain security.

Shared security, on the other hand, tries to achieve the same goals—defend against invalid state transitions, re-orgs, and censorship—by using the resources of a main network to secure one or more secondary networks. The main goals of shared security are to increase capital efficiency in blockchain networks without introducing new risks and to boost the defensive capabilities of new protocols. While Polkadot launched with (essentially) this exact design from the start with its Relay-parachain model, more and more protocols are moving towards this idea, including Cosmos and Avalanche.

However, shared security can be implemented in various ways:

1. **Simultaneous Participation**: Validators operate on both the primary and secondary networks simultaneously.
2. **Random Sampling**: A subset of validators from the primary network is randomly selected to secure the secondary network.
3. **Independent Validators**: The secondary network is secured by an independent set of validators bonded on the primary network.
4. **Re-delegation of Stakes**: Validators from the primary network re-delegate their staked capital to validators on the secondary network.

Regardless of the implementation, a key aspect of shared security is the ability of the "Primary" network to punish malicious behavior on the secondary network. If a validator on the secondary network is malicious, an honest participant can file a dispute by presenting evidence of this behavior to the "Primary" network. Acting as a judge, the "Primary" network verifies the evidence and punishes the dishonest validators if the claim is validated by slashing their collateral.

For slashing mechanisms to be effective, misbehavior must be attributable to specific parties. In PoS networks, this is achieved through the use of a unique cryptographic identity for each validator. Validators sign block data with their private keys during their duties, binding them to their actions. This setup enables the network to penalize validators for actions that threaten the network's safety or liveness, such as:

- **Equivocation**: Signing two conflicting blocks during the same period.
- **Signing Invalid Blocks**: Whether during a proposal or attestation.
- **Censorship**: Blocking or hiding transactions or parts of block data.

PART 9
PROMISING INTEROPERABILITY TECHNOLOGIES

Emerging interoperability technologies in the blockchain space, such as zero-knowledge proofs (ZKPs), intent-based systems, and modular scaling strategies like chain abstraction, are reshaping how different blockchains communicate and integrate. Many of these can be used to improve existing solutions like bridges and interoperability layers, while others usher in an entirely new paradigm. These innovations look to address the longstanding challenges of scalability, efficiency, and user experience within the blockchain interoperability space.

9.1 ZERO-KNOWLEDGE PROOFS

Zero-knowledge proofs (ZKPs) are a cryptographic technique allowing one party to prove knowledge of certain information without revealing it. This innovative method enhances blockchain security and efficiency by enabling the verification of off-chain computations on-chain, thus speeding up processes and reducing gas costs. ZKPs work by having a verifier challenge a prover with tasks that can only be accurately completed if the prover possesses the relevant knowledge. Various implementations of ZKPs, such as zk-SNARKs, zk-STARKs, PLONKs, and Bulletproofs, each offer different advantages regarding proof size, verification speed, and resource efficiency.

In the context of blockchain interoperability, ZKPs, particularly zk-SNARKs, are key as they move verification from human validators to cryptographic methods. This places the trust in code, not humans, and eliminates the chance of human malicious interference. This allows for trustless, secure transactions across different blockchains that have been designed to interoperate in this fashion.

In this model, one or more parties generate SNARK proofs that attest to the validity of a blockchain's state. These proofs can be derived from elements such as block headers and can be verified by any number of participants without requiring interaction between them, which is a key advantage over interactive fraud proofs. The soundness of the SNARK system ensures that it is nearly impossible for an adversary to create a valid proof for an invalid state, making this approach highly secure.

Unlike staking-based systems, which require validators to lock up tokens as collateral, SNARK-based systems do not involve any staking or bonding mechanisms. This eliminates the inefficiency associated with capital being tied up for extended periods. Moreover, relayers are not required to post a bond before making claims about cross-chain transactions, as the SNARK proof itself serves as the verification.

SNARKs enable quick execution of cross-chain operations, as there is no need for delay periods to allow for fraud proofs, which are necessary for some other security models. Once a SNARK proof is verified, the cross-chain transaction can proceed immediately. However, it's important to note that generating SNARK proofs is computationally intensive, which could impact the system's efficiency compared to externally verified systems.

9.2 INTENTS

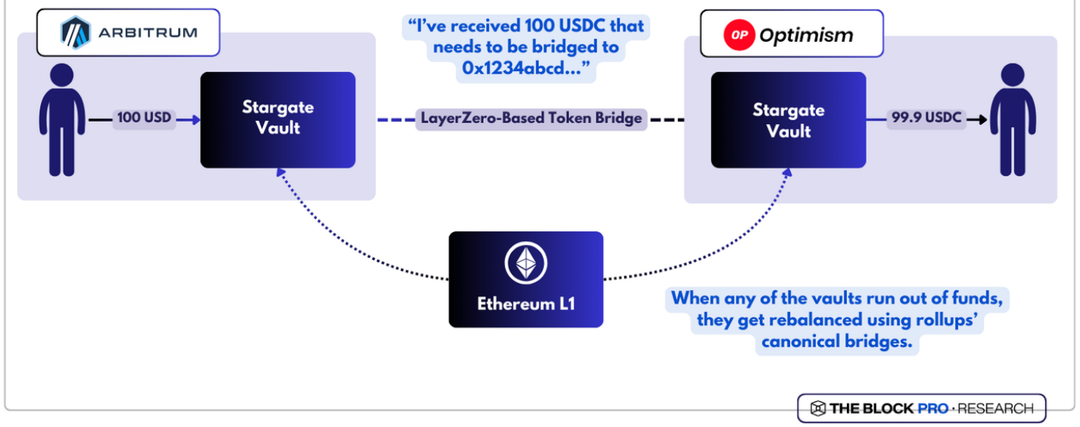
> "If you look at the two properties (that users care most about), they are speed and better price. Intent-based bridges, by eliminating the pools (from liquidity bridges) and simply fronting the capital, optimize for speed and offer a fast user experience."
>
> -Arjun Chand, Li.finance

In the blockchain world, the concept of an "intent" represents a user's desired outcome without specifying the exact steps to achieve it. This differs from traditional blockchain transactions, which require detailed instructions on executing the code's logic step-by-step. For example, a traditional transaction might specify a series of actions and exact amounts to achieve a goal. In contrast, an intent-based approach would simply communicate the user's ultimate desired outcome and be free to service that in any number of ways. This abstraction allows for more flexibility and efficiency, separating the "what" from the "how."

Traditional cross-chain bridges use messaging protocols to transfer tokens between chains. For example, Stargate uses LayerZero to send deposit information across chains, with the destination chain unlocking the corresponding tokens based on this message (Figure 25). The process essentially locks tokens on the source chain and sends a trust-based message to release them on the destination chain.

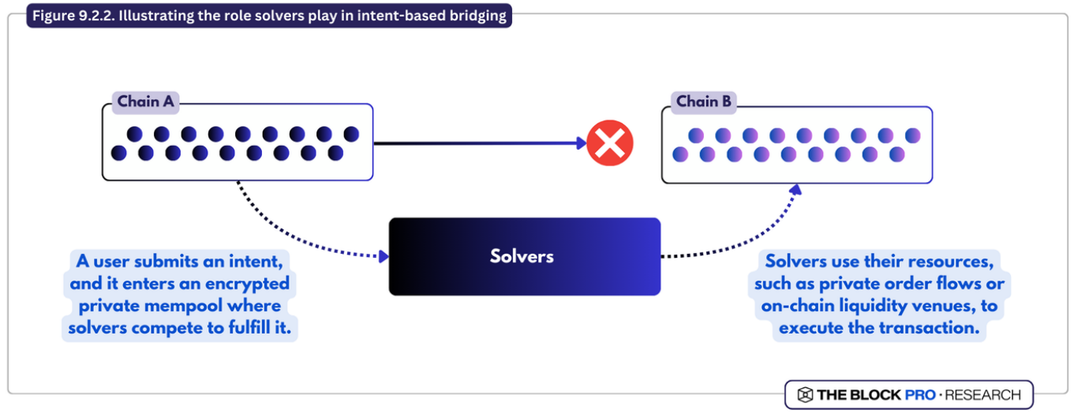
Specialized service providers called solvers handle the process of fulfilling these intents. Solvers are sophisticated entities that determine the most efficient way to achieve the user's goal (Figure 26). They might utilize various decentralized exchanges (DEXes), break transactions into smaller parts, or employ other strategies to fulfill the user's intent in the best possible way. By optimizing the execution, solvers can often secure better prices and reduce slippage compared to traditional transaction methods. Protocols like CowSwap and Synapse exemplify this model.

Figure 9.2.1. A simplified diagram of the LayerZero-based token bridge, Stargate



Source: Stargate

Figure 9.2.2. Illustrating the role solvers play in intent-based bridging



Source

When a user submits an intent, it enters an encrypted private mempool where solvers compete to fulfill it. Solvers use their resources, such as private order flows or on-chain liquidity venues like Uniswap and Curve, to execute the transaction. The competition among solvers drives down margins so users get the best possible execution. This abstracts away the complexity of blockchain interactions and makes the user experience seamless and efficient.

The key issue for many bridge designs is how the architectures address cost and speed, which are crucial factors in bridging. Users prioritize efficiency and returns, often relying on aggregators to choose the best bridge. Intents have a clear advantage in speed over general message passing since they process the transaction(s) offchain and only settle onchain when necessary. In contrast, messages are limited by the finality time of transactions. Cost-effectiveness in these systems depends on two factors: the mechanism for verification and the ability to scale liquidity. Intents benefit from optimistic verification, reducing costs by eliminating the need for messages.

An emerging component of implementing intents is the use of zero-knowledge proofs (discussed in the previous section). In the context of intents, ZK-proofs can validate transactions and ensure compliance with the terms set by the involved parties, all while preserving the privacy of the underlying data.

The benefits of using intents include:

1. **Enhanced Privacy**: With zero-knowledge proofs, intents allow users to engage in transactions without exposing sensitive or personal information, safeguarding user privacy and security.
2. **Scalability**: Intents can significantly reduce the burden on the blockchain by processing transactions off-chain and settling them on-chain only when necessary. This method decreases network congestion and increases transaction throughput.

While the advantages of intents are clear, there are several challenges that need addressing:

- **Complexity in Implementation**: The technical sophistication required to implement intents and zero-knowledge proofs can be a barrier for many organizations without the necessary expertise.
- **Scalability Trade-offs**: Although intents help in reducing load on the blockchain, the computational complexity involved in zero-knowledge proofs might still lead to scalability issues under certain conditions.

Ultimately, intent-based systems, supported by solvers, pave the way for a more intuitive and user-friendly blockchain experience, making it easier for users to achieve their goals without needing to navigate the complex details of blockchain technology. This approach is a key step toward realizing the vision of chain abstraction, where users can seamlessly interact with various blockchains without concern for the underlying mechanics.

9.3 CHAIN ABSTRACTION

With a few notable exceptions, Solana being the primary example, the problem of blockchain scalability has moved towards a modular approach, separating the different functional layers of a blockchain, such as settlement, data availability, and execution. While this method has spurred the development of solutions like L2s, rollups, data availability layers, sidechains, and state channels, it has also led to a fragmented user experience, making it difficult for users to navigate across different platforms.

With this modular scaling strategy, gone are the days when a single blockchain could dominate the entire ecosystem. Instead, crypto's future is looking increasingly multi-chain by the year. The complexity and diversity of today's blockchain modular solutions and the multi-chain world have made it necessary to rethink how users interact with this technology. Enter chain abstraction—an innovative approach designed to unify the fractured modular landscape of Web3 by abstracting away the complexities of different blockchains. Chain abstraction aims to enable seamless interaction across multiple blockchains without requiring users to understand or manage the underlying technologies.

> "I think we're in the world where all the people that could or want to understand the backend infrastructure (of interoperability protocols) have done that, and then the rest of the people do not care."
>
> - Sergey Gorbunov, Axelar

Inspired by Account Abstraction (AA) in Ethereum, in which user accounts and smart contracts are combined into a single account type, chain abstraction aims to simplify the user's multi-chain experience.

It allows dApps to execute logic across any blockchain without users having to switch networks or manage multiple wallets. This means that users can interact with dApps using any supported token from any chain within a single, consistent interface without needing to navigate between different platforms.

One key feature of chain abstraction is its ability to manage gas and transaction details within the abstraction layer, freeing users from acquiring or manually spending gas on secondary chains. This not only simplifies the user experience but also lowers the entry barrier for average users, making blockchain technology more accessible to a broader audience.

One of the key innovations to chain abstraction is the deployment of ZKPs, which allow concise proofs to validate transactions across multiple chains. Instead of processing the entire transaction on-chain, only a proof is submitted, which reduces the data and processing required. Besides cost reduction, ZKPs also play a crucial role in protecting user privacy across multiple chains by keeping transaction details private across all platforms.

Additionally, intents play a crucial role in the chain abstraction movement by enabling users to interact with blockchain applications without dealing with the intricacies of the underlying technology. In an account-abstracted system, users can simply sign a transaction, and the execution is outsourced to solvers. These solvers manage the risks and complexities across different blockchains and applications, ensuring the user's intent is optimally fulfilled. By pricing their services according to the complexity and risks involved, solvers further simplify the user experience by handling gas fees and execution risks, aligning perfectly with the goals of chain abstraction.

The NEAR Protocol is a pioneer in the chain abstraction movement, actively developing solutions that enhance user experience across various blockchains. NEAR's efforts include account aggregation, Data Availability layers, intent brokers, decentralized frontends, and super wallet development. These innovations allow users to seamlessly engage with platforms like Ethereum and Avalanche using a single NEAR account, streamlining interactions and contributing to the broader adoption of Web3 technologies.

Another intent-based protocol, Across, is a cross-chain bridge protocol that facilitates fast and affordable transactions by leveraging an optimistic oracle, bonded relayers, and single-sided liquidity pools. The launch of Across V3 introduces an innovative intent order structure, enabling seamless development of single-chain dApps that support cross-chain operations. V3's composable intents engine consists of three key components: a request-for-quote (RFQ) system for intent-based orders, a network of third-party relayers with off-chain liquidity, and an optimistic verification-based settlement mechanism. This modular architecture allows developers to connect to specific parts of the stack as needed. The protocol optimizes cost efficiency by aggregating multiple settlements into a single message, reducing gas fees and the overall complexity of on-chain verification.

9.4 L2 AGGREGATION LAYERS AND INTEROPERABILITY

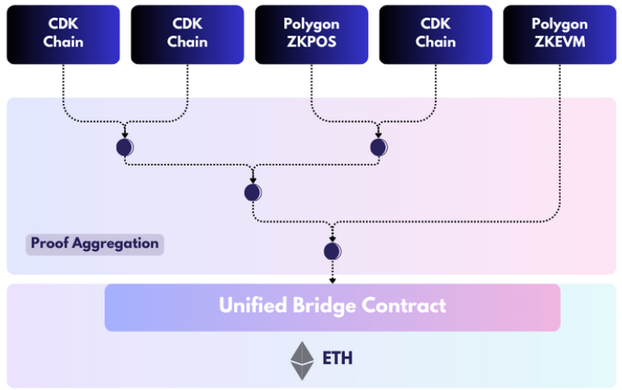
As more activity moves to L2s and rollups, the same interoperability challenges persist but with new and unique constraints. This has led to the creation of various solutions to create a more cohesive and efficient blockchain ecosystem across the L2s. Two of the notable ones are Polygon's Aggregation Layer (AggLayer) and Optimism's Superchain, both with their own approaches to solving this problem.

Polygon's Aggregation Layer's (AggLayer) core innovation lies in its ability to unify disparate L2 networks into a cohesive framework, enabling seamless asset and data transfers across platforms. By introducing a "layer of layers," AggLayer seeks to enhance the interoperability of L2s that utilize the Polygon CDK while preserving the autonomy of individual blockchains.

The AggLayer aggregates the proofs and state updates from L2 chains and submits them to Ethereum (Figure 27). The design is structured around three main components: proof aggregation, optimistic batch confirmation, and atomic cross-chain interaction. The optimistic confirmation mechanism solves the latency problem by allowing batches to be pre-confirmed based on message validity, so cross-chain messaging is faster.

Validity in this context is maintained through a process where chains can submit batches and message queues without immediate proof, relying on AggLayer to confirm the pre-confirmed state. This approach safeguards against inconsistencies and ensures the integrity of cross-chain transactions.

Figure 9.4.1. Polygon's AggLayer

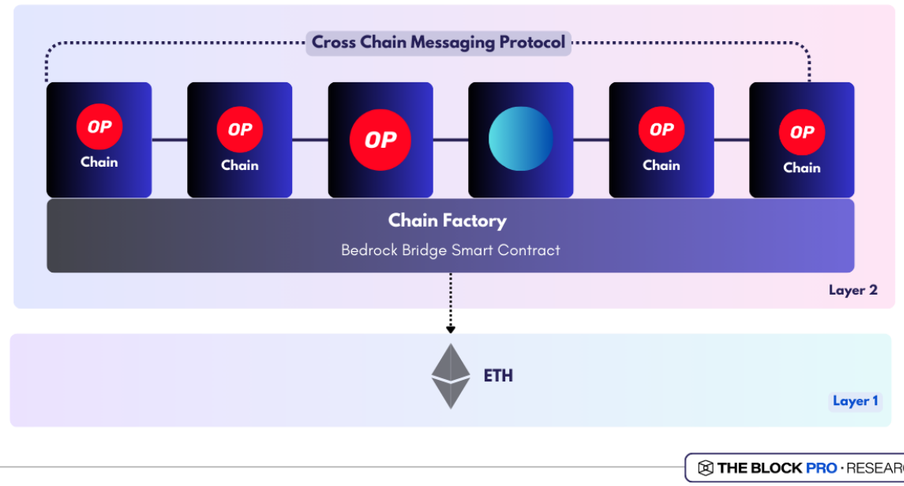


Source

A key feature of AggLayer is its emphasis on economic sovereignty, allowing each blockchain to maintain its governance and operational independence. This sovereignty is coupled with shared liquidity and interoperable capabilities, which collectively foster a more interconnected and efficient ecosystem. The AggLayer utilizes ZKPs to validate transactions across different chains and includes a decentralized protocol that aggregates liquidity and streamlines the execution of smart contracts. This flexibility allows various blockchain projects to optimize for speed, transaction costs, or specific use-case requirements. By democratizing access to liquidity, AggLayer lowers barriers to entry for new projects, enhancing the economic viability of smaller networks and promoting a more competitive environment.

Optimism's Superchain initiative (Figure 28), while sharing some conceptual similarities with AggLayer, adopts a different approach to blockchain interoperability. Built using the Optimistic Stack, the Superchain aims to interconnect Ethereum-aligned L2 chains through atomic cross-chain composability, decentralized governance, and a shared Ethereum infrastructure. This architecture seeks to create uniform blockspace and upgradability across networks, simplifying development and scaling efforts. Notable entities like Base, Worldcoin, Zora, and Debank have already adopted or plan to adopt the OP Chain blueprint.

Figure 9.4.2. Optimism's Superchain stack



Source

A significant challenge that the Superchain addresses is the current reliance on Ethereum L1 for secure communication and asset movement between chains. This reliance, however, comes with high costs and slow transaction speeds, leading to a fragmented ecosystem. The Superchain's architecture attempts to mitigate these issues by standardizing the development model for OP Chains, allowing developers to focus on building within the Superchain without needing to manage individual chains' complexities.

One of the critical components of the Superchain is shared sequencing technology, which enables a sequencer to build a batch for multiple rollups. This technology is vital for maintaining atomicity in cross-chain interactions, ensuring that any invalid transaction within a batch can cause the entire batch to be disputed and reverted. However, the optimistic design of the Superchain limits interoperability with other rollups outside of its ecosystem, which could pose challenges as the ecosystem evolves. Other key properties of the Superchain system are illustrated in Figure 29 below.

Figure 9.4.3. Highlighting key properties of the Superchain

Property	Purpose
Shared L1 Blockchain	General-Provides a total ordering of transactions across all OP Chains
Shared Bridge for all OP Chains	Enables OP Chains to have standardized security properties
Cheap OP Chain Deployment	Enables deploying and transacting on OP Chains without the high fees of transacting on L1
Configuration Options for OP Chains	Enables OP Chains to configure their data availability. provider, sequencer address, etc.
Secure Transactions and Cross Chain Messages	Enables users to safely migrate state between OP Chains

THE BLOCK PRO · RESEARCH

Source: Binance Research

Despite its potential, the Superchain faces concerns regarding centralization, particularly with the shared sequencing model. This approach risks creating oligopolistic control within the rollup ecosystem, potentially stifling innovation and leading to a fragmented Ethereum ecosystem. New rollups may find themselves at a crossroads: either integrate into the existing stack and sacrifice broader interoperability or explore alternatives such as zero-knowledge (ZK) technology.

CONCLUSION

Cross-chain interoperability is transforming the blockchain landscape by enabling diverse networks to communicate, share data, and exchange value. This report has highlighted the importance of bridging solutions, general message passing, and shared security in achieving a more connected blockchain ecosystem. Despite challenges like security risks, validator trust, and technical complexity, the space has seen considerable advancements, from bridge maturation to intent-based protocols and zero-knowledge proof (ZKP) mechanisms.

As of 2024, cross-chain bridges and communication protocols remain foundational for blockchain interoperability. While the TVL locked in bridges represents only a fraction of the broader DeFi market, the still significant funds underscore the growing importance of these solutions and the need to protect them. The technology behind interoperability, such as LayerZero's general message passing and Polkadot's XCM, is paving the way for more fluid cross-chain interactions, which are crucial for building an interconnected multi-chain ecosystem.

The future of blockchain interoperability lies in innovations such as intent-based systems, ZKPs, and inter-chain messaging protocols like IBC and XCM. These technologies aim to simplify user experiences, improve transaction efficiency, and reduce trust dependencies. As the blockchain ecosystem becomes more multi-chain-oriented, these emerging solutions are likely to grow in significance but potentially fade into the background as solutions like chain abstraction and others obscure more and more of the "backend" infrastructure powering the crypto ecosystem.

DISCLOSURES

This report is sponsored by The Web 3 Foundation. The content of this report contains views and opinions expressed by The Block's analysts which are solely their own opinions, and do not necessarily reflect the opinions of The Block or the organization that commissioned the report. The Block's analysts may have holdings in the assets discussed in this report and this statement is to disclose any perceived conflict of interest. Please refer to The Block's Financial Disclosures page for author holdings.

Beginning in 2021, Michael McCaffrey, the former CEO and majority owner of The Block, took a series of loans from founder and former FTX and Alameda CEO Sam Bankman-Fried. McCaffrey resigned from the company in December 2022 after failing to disclose those transactions.

This report is for informational purposes only and should not be relied upon as a basis for investment decisions, nor is it offered or intended to be used as legal, tax, investment, financial or other advice. You should conduct your own research and consult independent counsel on the matters discussed within this report. Past performance of any asset is not indicative of future results.