Cross-Chain Transfer Protocol (CCTP) V2

Walker Mayerchak Mike Grant Jonathan Lim

Chase McDermott Elim Poon Adrian Soghoian

Kaili Wang Gordon Y. Liao

March 2025

Abstract

CCTP is a cross-chain, permissionless, general-purpose messaging protocol optimized for stablecoin transfers. CCTP V2 significantly improves upon V1 by reducing transfer times from slow-finality chains like Ethereum from approximately 15 minutes to mere seconds.[1] This reduction is achieved through an off-chain global allowance mechanism with an over-collateralization pool that enables message attestation before source-chain transaction finalization. By eliminating the need for chain-specific liquidity pools, CCTP V2 enhances capital efficiency while maintaining security. Users can transfer native stablecoins cross-chain nearly instantly without introducing additional trust assumptions beyond Circle as the stablecoin issuer, offering a secure and efficient solution to blockchain interoperability challenges.

1 Introduction

The proliferation of blockchain ecosystems has created a fragmented landscape for digital assets, particularly stablecoins. This fragmentation presents significant challenges for liquidity efficiency, interoperability, and systemic risk management—issues that have drawn increasing attention from market participants and global regulatory bodies alike [Committee on Payments and Market Infrastructures, 2023, Boissay et al., 2022].

In the current multi-chain environment, stablecoin transfers between blockchains typically rely on one of two approaches: (1) lock-and-mint bridges that create wrapped, non-native versions of stablecoins, or (2) market maker/intent-based solutions that depend on pre-funded reserves on each chain. Both approaches introduce significant inefficiencies and risks. Lock-and-mint bridges have been particularly vulnerable to security exploits, with over $2 billion lost to bridge hacks between 2021 and 2022 according to industry analyses [Chainalysis, 2022].

Meanwhile, market maker/intent-based solutions fragment capital efficiency by requiring separate pools of the same asset across different chains. This fragmentation creates inconsistent user experiences, reduces overall market depth, and increases costs as liquidity providers charge fees to compensate for capital lockup and risk.

Circle's Cross-Chain Transfer Protocol (CCTP) offers a fundamentally different approach that provides both general-purpose message-passing and native value-transfer capabilities. By leveraging Circle's position as the stablecoin issuer, CCTP avoids the common risks of traditional lock-and-mint bridges. Instead of holding stablecoins in escrow, CCTP employs a burn-and-mint model, destroying tokens on the source blockchain and minting new, equivalent native tokens on the destination chain. This approach preserves the fungibility and integrity of the asset across chains without fragmenting liquidity.

CCTP version 2 significantly enhances this model by introducing smart contract interfaces that support "Fast Messages"—messages that are signed and attested by off-chain services before the block containing the originating message is finalized on the source blockchain. It also introduces an over-collateralization pool, as a part of the issuer's capital, that makes up a global allowance shared across chains to guard against source chain re-organizations and potential unbacked mints.[2] These innovations enable stablecoin transfers between blockchains in seconds—even before source-chain transactions are fully finalized while effectively managing risks associated with blockchain reorganizations.

Furthermore, CCTP V2 also introduces *Hooks*, which enable arbitrary actions to be executed atomically with stablecoin transfers without additional trust assumptions, thereby deepening composability with smart contracts.

By combining the advantages of faster-than-finality cross-chain messaging and shared over-collateralization pool, CCTP V2 addresses the central liquidity and interoperability challenges highlighted by global payment standard-setters [Committee on Payments and Market Infrastructures, 2023]. Its technical design mitigates security risks inherent in cross-chain bridging solutions, enhances capital efficiency, and streamlines cross-chain stablecoin transfers, offering an economically efficient and technically robust foundation for interoperable on-chain payments.

[1] Fast attestations in CCTP V2 are expected after approximately 16 seconds for messages from Ethereum L1, and 4 seconds for messages from Ethereum L2s.

[2] For discussion on stablecoin capital framework, see Liao et al. [2024].

2 Background

2.1 Finalization Times

CCTP V1 relies on blockchain finality to determine whether a valid burn message can be attested. Blockchain finality is crucial to ensure that a burn event remains in the chain long-term and is not forked away, which could otherwise lead to an "unbacked" stablecoin mint on the destination chain.

However, waiting for full finality can take significant time, especially for Ethereum mainnet and Layer 2 networks anchored to Ethereum consensus.

Table 1: CCTP V1 Required Condition Before Attesting

Source Chain	Condition	Average Time
Ethereum	ETH L1 Finality (65–94 slots)	∼15 minutes
Avalanche	1 block	a few seconds**
OP Mainnet	ETH L1 Finality*	∼15–20 min*
Arbitrum	ETH L1 Finality*	∼15–20 min*
Noble	1 block	a few seconds**
Base	ETH L1 Finality*	∼15–20 min*
Polygon PoS	∼200 blocks	∼8 minutes
Solana	32 slots	a few seconds**
Sui	1 block	a few seconds**
Aptos	1 block	a few seconds**

* For optimistic rollups anchored to Ethereum, finalization occurs once an L2 transaction batch is finalized on Ethereum.
** Chains labeled "a few seconds" exhibit sub-second finality and should generally be attested within a few seconds.

Waiting for full finality limits the utility of CCTP for applications that require lower latency. CCTP V2 addresses this by introducing Fast Transfers, which can move stablecoins without waiting for full finality.

2.2 System Architecture

CCTP V2, like V1, supports arbitrary cross-chain messages. Stablecoin transfers are one application of this general-purpose messaging system, facilitated by the TokenMessenger smart contracts and the BurnMessage integrated with the MessageTransmitter. See Figure 1.

Circle's off-chain attestation service, Iris, observes the messages emitted by the MessageTransmitter and attest to them once the block containing the message reaches a required finality threshold. The attestation consists of a set of signatures provided by the attesters. Iris exposes a permissionless REST API to retrieve message attestations for relaying to the destination chain.

CCTP V2 introduces a new set of smart contracts and message formats better suited to handle messages faster than blockchain finality, forming a network distinct from CCTP V1.

3 Model for Fast Transfer Allowance

This section presents a model for analyzing Fast Transfer usage in CCTP V2. The model centers on how the global Fast Transfer allowance—the maximum amount of tokens that can be minted before source-chain finality—influences fee determination, user adoption, and capital efficiency across multiple blockchains with varying characteristics. This framework provides a foundation for understanding how CCTP V2's Fast Transfer allowance balances speed, risk, and capital efficiency in cross-chain stablecoin transfers.

3.1 Model Setup

Consider n source chains indexed by $c = 1, \ldots, n$. Each chain c is characterized by:

- A finality time T_c (e.g., ∼15 minutes for Ethereum, a few seconds for Avalanche).
- A reorg risk factor p_c, representing the probability or severity of reorganization events.
- A user arrival rate λ_c of potential transfers per unit time.
- A distribution of transfer sizes $F_{x|c}(x)$, where x represents the amount of stablecoin for each individual transfer.
- A distribution of time-sensitivity $F_{v|c}(v)$, representing users' willingness to pay for Fast Transfer.

Users on chain c choose Fast Transfer if their time-sensitivity $v \geq f_c$, where f_c is the chain-specific fee. Otherwise, they select the slower standard transfer option. This choice mechanism reflects how CCTP V2 allows users to signal their preference for speed through their willingness to pay a fee, creating a natural market-based allocation of the limited fast transfer allowance.

3.2 Global Allowance and In-Flight Volume

All chains share the same global allowance A, which represents Circle's over-collateralization to cover potential reorg risks. At any time, Circle must ensure that the sum of in-flight fast transfers across all chains does not exceed A.

The fraction of users on chain c who opt for Fast Transfer is:

$$\alpha_c = \Pr(v \geq f_c) = 1 - F_{v|c}(f_c)$$

The expected volume of in-flight transfers on chain c is:

$$V_c = \alpha_c \, \lambda_c \, \mathbb{E}[\, x \mid v \geq f_c \,] \, T_c$$

The global allowance constraint is then:

$$\sum_{c=1}^{n} V_c = \sum_{c=1}^{n} (\alpha_c \, \lambda_c \, \mathbb{E}[\, x \mid v \geq f_c \,] \, T_c) \leq A$$

This constraint ensures that the total volume of tokens at risk (those minted on destination chains before source chain finality) never exceeds the allowance A. Unlike

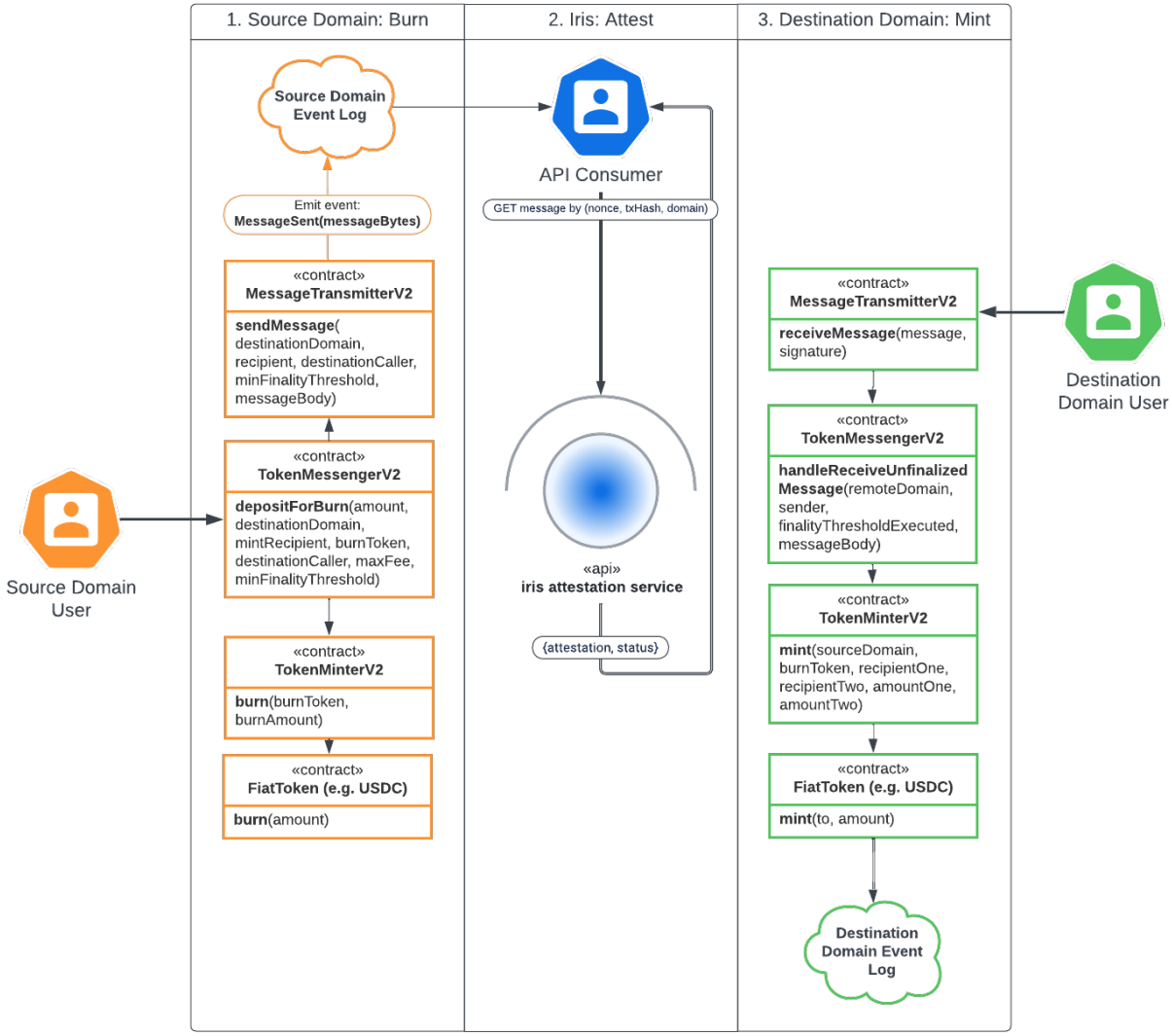

Figure 1: System Architecture Diagram

traditional bridge designs that require separate liquidity pools for each chain, CCTP V2's global allowance approach enables more efficient capital utilization by allowing the same collateral to support fast transfers across multiple chains simultaneously.

3.3 Chain-Specific Fees and Risk Premium

Because each chain has different T_c and p_c, fees f_c for each chain can be different. A chain with longer T_c or higher p_c creates higher expected losses or higher cost of capital, implying a higher Fast Transfer fee. Formally:

$$f_c = g(\lambda_c, T_c, p_c, A, \dots)$$

where g is an increasing function of T_c and p_c.

The fee f_c can be decomposed into two components:

$$f_c = r_c + s_c$$

where r_c represents the risk premium associated with chain c's reorg probability, and s_c represents the scarcity premium reflecting the opportunity cost of consuming the shared allowance. This fee structure aligns with CCTP V2's design principles of risk-based pricing and efficient allowance allocation.

3.4 Equilibrium and Implications

In equilibrium, Circle sets fees $\{f_1, f_2, \dots, f_n\}$ to maximize some objective function (e.g. volume, or user welfare) subject to the allowance constraint. The model yields several key properties and insights for CCTP V2:

- **Fee Differentiation:** Chains with longer finality times (higher T_c) or higher reorg risk (higher p_c) face higher fees, as they consume allowance longer and pose greater risk. During high demand, the scarcity premium component increases to ration the limited allowance.
- **Capital Efficiency:** The global allowance approach is more capital-efficient than chain-specific liquidity pools, as it allows the same collateral to support fast transfers across multiple chains.
- **Crowding Out Effects:** Because the allowance is global, high usage on one chain can reduce available

allowance for others. During demand surges, some chains may experience higher fees or reduced fast transfer availability.

- **Dynamic Fee Adjustment:** In practice, fees may need to adjust dynamically based on current allowance utilization, creating a market-based mechanism for allowance allocation.
- **Guaranteed Allowance Potential:** Future extensions could include reserved portions of the allowance for specific users or chains, potentially with premium pricing for guaranteed availability.

This model demonstrates how CCTP V2's fast transfer allowance mechanism creates a more efficient cross-chain transfer system compared to traditional bridging solutions. By using a global allowance rather than chain-specific liquidity pools, CCTP V2 achieves greater capital efficiency while maintaining security against reorganization risks. The fee structure naturally allocates the scarce allowance resource to users with the highest time-sensitivity, optimizing the overall utility of the system.

In more advanced implementations, the model could be extended to incorporate dynamic fee adjustments based on real-time allowance utilization, creating a market-based mechanism that responds to changing demand patterns across different chains. This would further enhance the efficiency of allowance allocation and improve the user experience during periods of high demand.

4 Fast Transfer Allowance

4.1 Managing Re-Org Risk

CCTP V2 improves the user and developer experience by reducing latency for bridging. However, this improvement comes with risks related to pre-finality attestations, especially for fiat-backed stablecoins like USDC.

A major risk is that a Fast Transfer attestation may be published, but then the source chain event is reorganized or re-ordered, resulting in a mint on the destination chain without a corresponding burn on the source chain. This risk is minimized by calibrating when attestations are provided (i.e., attesting only when re-organizations are statistically very unlikely), though not entirely eliminated.

Since only full blockchain finality guarantees reorganization resistance, CCTP V2 introduces the Fast Transfer Allowance construct, which constrains the maximum amount of in-flight Fast Transfers across the network.

4.2 Allowance Design

The Fast Transfer Allowance in CCTP V2 is an over-collateralization pool funded by the stablecoin issuer. It serves as a bookkeeping tool to limit how much stablecoin can be transferred via Fast Transfers at any time.

When Iris attests to a Fast Transfer, it consumes an equivalent amount from the Fast Transfer Allowance. In contrast, Standard Transfers (which await finality) do not consume the Allowance.

The Allowance can be replenished in one of three ways:

(1) When the block containing a burn event for a Fast Transfer is eventually finalized on the source chain (typically around 15 minutes for Ethereum).

(2) If the attestation has not been used to mint and the destination chain's encoded `expirationBlock` is reached and finalized, the allowance is automatically replenished.

(3) Circle supplies additional capital to over-collateralization pool.

The total allowance consumed equals the sum of approved, non-finalized, and non-expired Fast Transfer requests. The current Fast Transfer Allowance can be queried via the Iris API.

4.3 Allowance Allocation

As the Allowance is a finite resource shared across CCTP V2 Fast Transfer users, contention may arise. Iris balances fairness (processing transfers on a first-come, first-served basis) with availability (ensuring transfers are available to most users).

If there is insufficient allowance for a Fast Transfer, Iris will skip it and process subsequent transfers up to a limit, then retry the skipped transfer after a configurable delay. If the available allowance remains insufficient, the transfer will gracefully fall back to a Standard Transfer once the burn message reaches full finality.

5 Fast Transfer Expiration and Reattestation

5.1 Fast Transfer Expiration

The off-chain allowance reserved by each Fast Transfer is released once the burn either finalizes or expires. Before attestation, Iris encodes an `expirationBlock` approximately one hour in the future on the destination chain.[3] If the Fast Transfer expires, it must be re-attested with a finalized finality threshold. This expiration period balances mitigating stalled blockchain processing and giving relayers enough time to broadcast the attestation.

5.2 Re-Attestation

Destination chain message recipients may require specific finality thresholds (for example, only accepting finalized messages). To prevent messages from being stuck because the initial finality threshold was too low, messages can be re-attested with a higher threshold.[4] This re-attestation produces a new attestation with the same nonce but a higher `finalityThresholdExecuted`.

[3] One hour is selected as an initial compromise, but may be adjusted over time.

[4] Note that a fee will always be collected when allowance is consumed.

Initially, re-attestation is supported via the Iris API and can also be used to obtain a finalized attestation for an expired Fast Transfer.

6 Messaging Formats

CCTP V2 defines message formats for two types of cross-chain messages:

- **Message:** The general-purpose message envelope containing an arbitrary message body and metadata.
- **BurnMessage:** A stablecoin-specific transfer message that is embedded within a Message.

In V2, these messages include several additional fields.

Message Structure

Field	Bytes	Type	Idx*
version	4	uint32	0
sourceDomain	4	uint32	4
destinationDomain	4	uint32	8
nonce	32	bytes32	12
sender	32	bytes32	44
recipient	32	bytes32	76
destinationCaller	32	bytes32	108
minFinalityThreshold	4	uint32	140
finalityThresholdExecuted	4	uint32	144
messageBody	dyn	bytes	148

*The index indicates the byte offset position where decoding of each field begins.

`minFinalityThreshold` specifies the minimum speed at which Iris should attest (e.g., 1000 for fast messages, 2000 for standard), and `finalityThresholdExecuted` is set at attestation time.

BurnMessage Structure

Field	Bytes	Type	Idx
version	4	uint32	0
burnToken	32	bytes32	4
mintRecipient	32	bytes32	36
amount	32	uint256	68
messageSender	32	bytes32	100
maxFee	32	uint256	132
feeExecuted	32	uint256	164
expirationBlock	32	uint256	196
hookData	dyn	bytes	228

Here, `maxFee` represents the maximum fee the user is willing to pay for the transfer, while `feeExecuted` is the actual fee deducted from the bridged amount. `expirationBlock` is the destination chain block number at which the Fast Message attestation expires (set to 0 for Standard Transfers). `hookData` optionally carries additional metadata for Hooks.

7 Fee Collection

CCTP V2 supports fee collection to facilitate Fast Transfers that operate faster than blockchain finality. Circle may set a Fast Transfer fee for any CCTP V2 route, and these fees are published via the Iris API to ensure transparency and predictability.

The `maxFee` provided by the user, along with the `minFinalityThreshold` value, signals the intent for a Fast Transfer. If `maxFee` is not sufficiently high relative to the fee set in Iris API, the transfer is processed as a Standard Transfer. Fees are then deducted on the destination chain based on the `feeExecuted` value in the attestation message.

8 Finality Thresholds

CCTP V2 introduces the concept of a **Finality Threshold** which indicates the speed at which Iris processes the source blockchain event emission. This threshold differentiates between Fast (pre-finality) and Standard (post-finality) messages.

Table 2: Finality Threshold Values in CCTP V2

Threshold	Source Chain Status	Attestation Speed	Re-org Risk
1000	Pre-finality	Fast	Low
2000	Finality	Standard	None

If a developer specifies a `minFinalityThreshold` of 1000 and conditions for a Fast Transfer are met (e.g. sufficient allowance), Iris sets `finalityThresholdExecuted` to 1000. Otherwise, Iris waits for full finality and sets the value to 2000, ensuring a Standard Transfer. In the future, additional finality thresholds may be added.

9 CCTP V2 Nonces

In CCTP V1, each message was assigned a nonce (an auto-incrementing integer) for replay protection on the destination chain. However, with Fast Transfers and the possibility of re-organizations, this method could lead to conflicts if a message were re-orged after attestation.

CCTP V2 replaces this with a nonce computed off-chain at attestation time, using multiple inputs from the transaction and event emission. This nonce is more resistant to re-orgs and prevents duplicate assignments. Integrators can query the Iris API by transaction hash to retrieve all messages and corresponding attestations.

10 Trust-Minimized Composability with Hooks

In CCTP V1, third-party smart contracts integrated via wrapper contracts that mediated access to CCTP smart contracts and added features like fee collection. However, since metadata could not be embedded in stablecoin transfers, it had to be transmitted separately—introducing additional trust assumptions and complexity.

CCTP V2 addresses this by introducing Hooks that embed arbitrary data directly into each stablecoin transfer message. This metadata is attested along with the other BurnMessage V2 fields, reducing trust assumptions and enabling richer composability.

When combined with a destination chain wrapper contract, Hooks can enable:

- Encoding of relayer fee information for collection.
- Instructions to trigger a swap and deliver assets.
- Initiation of a swap in conjunction with another CCTP bridge.
- Splitting a transfer among multiple recipients.

The destination chain smart contract is responsible for interpreting the Hook data, processing the attestation, and executing the instructions atomically. Additionally, the smart contract can be set as the `destinationCaller` to ensure the Hook instructions are followed.

11 Potential Future Work

11.1 Guaranteed Allowance

To enhance predictability for integrators, a future iteration of CCTP might implement a system for guaranteed Fast Transfer Allowance reservations. Currently, the allowance is shared on a best-effort first-come, first-served basis, which can lead to delays during peak usage. A reservation system would allow developers or institutions to secure a portion of the allowance, ensuring their transactions are not delayed, with associated fees for the reservation.

11.2 Additional Finality Thresholds

While CCTP V2 currently defines two thresholds—Confirmed (1000) and Finalized (2000)—different blockchains and use cases might require additional granularity. For example, higher thresholds (greater than 2000) could be used for high-value or regulated transactions requiring extra security, whereas lower thresholds might suffice for applications needing even lower latency. A similar use-case for Ethereum "super finality" is described by Neu et al. [2024]. Conversely, even lower thresholds are possible, for applications with lower latency requirements.

References

Frederic Boissay, Giulio Cornelli, Sebastian Doerr, and Jon Frost. Blockchain scalability and the fragmentation of crypto. BIS Bulletin 56, Bank for International Settlements, June 2022. URL https://www.bis.org/publ/bisbull56.pdf.

Chainalysis. Cross-chain bridge hacks emerge as top security risk. August 2022. URL https://www.chainalysis.com/blog/cross-chain-bridge-hacks-2022/. Accessed: 2025-03-10.

Committee on Payments and Market Infrastructures. Considerations for the use of stablecoin arrangements in cross-border payments. Technical report, Bank for International Settlements, October 2023. URL https://www.bis.org/cpmi/publ/d220.pdf.

Gordon Liao, Dan Fishman, and Jeremy Fox-Geen. Risk-based capital for stable value tokens. *Available at SSRN*, 2024.

J. Neu, S. Sridhar, L. Yang, and D. Tse. Optimal flexible consensus and its application to ethereum. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 3885–3903, San Francisco, CA, USA, 2024. doi: 10.1109/SP54263.2024.00135.