Generalized Message Passing

CCTP architecture on EVM and non-EVM domains [Suggest Edits](#)

Cross-Chain Transfer Protocol (CCTP) uses generalized message passing to facilitate the native burning and minting of USDC across supported blockchains, also known as [domains](#) . Message passing is a three-step process:

1. An on-chain component on source domain emits a message.
2. Circle's off-chain attestation service signs the message.
3. The on-chain component at the destination domain receives the message, and forwards the message body to the specified recipient.

Architecture

On-chain components on all domains have the same purpose, but implementation differs between EVM-compatible and non-EVM domains.

CCTP on EVM Domains

The relationship between CCTP's on-chain components and Circle's off-chain attestation service is illustrated below for a burn-and-mint of USDC between EVM-compatible domains:

On EVM domains, the on-chain component for cross-chain burning and minting is called TokenMessenger , which is built on top of Message Transmitter , an on-chain component for generalized message passing.

In the diagram above, a token depositor calls the [TokenMessenger#depositForBurn](#) function to deposit a native token (such as USDC), which delegates to the TokenMinter contract to burn the token. The TokenMessenger contract then sends a message via the [MessageTransmitter#sendMessage](#) function. After [sufficient block confirmations](#) , Circle's off-chain attestation service, Iris, signs the message. An API consumer queries this attestation and submits it on-chain to the destination domain's [MessageTransmitter#receiveMessage](#) function. For more details, see [Quickstart: Cross-chain USDC transfer](#) .

To send an arbitrary message, directly call [MessageTransmitter#sendMessage](#) . Note that the message recipient must implement [IMessageHandler#handleReceiveMessage](#) . It is not currently possible to perform a burn-and-mint of USDC and add arbitrary data to the same message; arbitrary data must be included in a separate message.

CCTP on Non-EVM Domains

Noble

Noble is a Cosmos application-specific blockchain (or "appchain") that provides native asset issuance for the Cosmos ecosystem. USDC is natively issued on Noble and can be transferred via the Inter-Blockchain Communication (IBC) protocol to other supported appchains in Cosmos, or via CCTP to any supported domain (e.g. Ethereum).

Note that there are key differences between Cosmos appchains like Noble and EVM-compatible blockchains. Unlike on EVM domains where CCTP is a set of smart contracts, CCTP on Noble is a Cosmos SDK module, which is deployed by Noble governance and built into the Noble blockchain. Cosmos appchains can use IBC to build composable flows with CCTP on Noble. Updated3 months ago * [Table of Contents](#) * * [Architecture](#) * * * [CCTP on EVM Domains](#) * * * [CCTP on Non-EVM Domains](#)