

Initialize native USDC market cUSDCv3 on Arbitrum

Updated as of block [19319324](#) at 2/27/2024, 8:55:35 AM ET

- ID: 178
- Proposer: [0xC66e426404C742D81655A9D80Ce58fdbCE468A9](#)
- Start Block: 18000713 (8/26/2023, 2:44:35 PM ET)
- End Block: 18020423 (8/29/2023, 8:58:47 AM ET)
- Targets: [0x4Dbd4fc535Ac27206064B68FfCf827b0A60BAB3f](#) ;
[0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48](#) ;
[0xBd3fa81B58Ba92a82136038B25aDec7066af3155](#) ;
[0x4976fb03C32e5B8cfe2b6cCB31c09Ba78EBaBa41](#)

Forum Post

Forum post is present here: [Forum Post](#)

Table of contents

- [Proposal Text](#)
- [Checks](#)
 - [Checks Compound Proposal Details](#)  [Passed](#)

Proposal Text

Initialize native USDC market cUSDCv3 on Arbitrum

This proposal takes the governance steps recommended and necessary to initialize a Compound III USDC (native USDC on Arbitrum) market on Arbitrum; upon execution, cUSDCv3 will be ready for use. Simulations have confirmed the market's readiness, as much as possible, using the [Comet scenario suite](#). Although real tests have also been run over the Goerli/Arbitrum Goerli, this will be the first proposal to mint native USDC on Arbitrum mainnet by burning USDC on mainnet via the `depositAndBurn` function on the Cross-Chain Transfer Protocol (CCTP) provided by Circle, and therefore includes risks not present in previous proposals.

The proposal sets the entire configuration in the Configurator to be the same as the existing bridged USDC.e market. Finally, the parameters include a migration of bridged USDC.e market supply-side COMP incentives to users in the new native USDC market.

Further detailed information can be found on the corresponding [proposal pull request](#) and [forum discussion](#).

Proposal Actions

The first proposal action sets the Comet Factory, Comet configuration and deploys a new Comet implementation on Arbitrum. This sends the encoded `setFactory`, `setConfiguration` and `deployAndUpgradeTo` calls across the bridge to the governance receiver on Arbitrum. It also calls `setRewardConfig` on the Arbitrum rewards contract, to establish Arbitrum's bridged version of COMP as the reward token for the deployment and set the initial supply speed to be 10 COMP/day and borrow speed to be 0 COMP/day. It calls another `setBaseTrackingSupplySpeed` and `setBaseTrackingBorrowSpeed` to set the supply speed and borrow speed of the existing USDC.e market to be 0 COMP/day. Lastly it calls `deployAndUpgradeTo` to deploy an

updated Comet implementation for the existing bridged USDC.e market to have the new supply and borrow rewards speed.

The second action approves Circle's Cross-Chain Transfer Protocol (CCTP) [TokenMessenger](#) to take the Timelock's USDC on Mainnet, in order to seed the market reserves through the CCTP.

The third action deposits and burns 10K USDC from mainnet via `depositForBurn` function on CCTP's [TokenMessenger](#) contract to mint native USDC to Comet on Arbitrum.

The fourth action updates the ENS TXT record `v3-official-markets` on `v3-additional-grants.compound-community-licenses.eth`, updating the official markets JSON to include the new native USDC market and renames the bridged USDC market's `baseSymbol` to `USDC.e` from `USDC`.


Checks

Checks Compound Proposal Details Passed

Info:

1- Bridge wrapped actions to Arbitrum

a-  Set factory of [USDC](#) to [CometFactory](#).

b-  Set configuration for [USDC](#) to:

```
{ governor: Timelock, pauseGuardian: GnosisSafeL2, baseToken: USDC, baseTokenPriceFeed: PriceFeed,
extensionDelegate: cUSDCv3, supplyKink: 0.000000000000000008, supplyPerYearInterestRateSlopeLow: 0.0325,
supplyPerYearInterestRateSlopeHigh: 0.4, supplyPerYearInterestRateBase: 0, borrowKink: 0.000000000000000008,
borrowPerYearInterestRateSlopeLow: 0.035000000000000004, borrowPerYearInterestRateSlopeHigh: 0.25,
borrowPerYearInterestRateBase: 0.015, storeFrontPriceFactor: 0.8, trackingIndexScale: 1,000,000,000,000,000,
baseTrackingSupplySpeed: 0.00000011574074074, baseTrackingBorrowSpeed: 0, baseMinForRewards: 0.000000001,
baseBorrowMin: 0.0000000000000000001, targetReserves: 0.000005, assetConfigs: [ { "asset": "ARB", "priceFeed":
"0xb2A824043730FE05F3DA2efaFa1CBbe83fa548D6", "decimals": "18", "borrowCollateralFactor":
"0.5500000000000000064", "liquidateCollateralFactor": "0.6", "liquidationFactor": "0.93", "supplyCap": "4000000" }, {
"asset": "GMX", "priceFeed": "0xDB98056FecFff59D032aB628337A4887110df3dB", "decimals": "18",
"borrowCollateralFactor": "0.4", "liquidateCollateralFactor": "0.45", "liquidationFactor": "0.93", "supplyCap": "50000" }, {
"asset": "WETH", "priceFeed": "0x639Fe6ab55C921f74e7fac1ee960C0B6293ba612", "decimals": "18",
"borrowCollateralFactor": "0.78", "liquidateCollateralFactor": "0.85", "liquidationFactor": "0.95", "supplyCap": "5000" }, {
"asset": "WBTC", "priceFeed": "0xd0C7101eACbB49F3deCcC166d238410D6D46d57", "decimals": "8",
"borrowCollateralFactor": "7000000000", "liquidateCollateralFactor": "7700000000", "liquidationFactor":
"9500000000", "supplyCap": "300" } ] }
```

c- Deploy and upgrade new implementation for [USDC](#) via [Configurator](#).

d-  Set reward token for market [USDC](#) as [COMP](#).

e- Set `BaseTrackingSupplySpeed` of [USDC](#) from 0 to 0 (It remains the same)

f- Set `BaseTrackingBorrowSpeed` of [USDC](#) from 0 to 0 (It remains the same)

g- Deploy and upgrade new implementation for [USDC](#) via [Configurator](#).

2-  Approve **10,000** [USDC](#) tokens to [0xBd3fa81B58Ba92a82136038B25aDec7066af3155](#)

3- Set `DepositforBurn` of [TokenMessenger](#) for the Burn contract [USDC](#) with amount 10,000, destination domain 3 and the Mint recipient `0x9c4ec768c28520b50860ea7a15bd7213a9ff58bf`

4- Set ENS text for v3-additional-grants.compound-community-licenses.eth with key: v3-official-markets and value:

```
{
  "1": [
    {
      "baseSymbol": "USDC",
      "cometAddress": "0xc3d688B66703497DAA19211EEdffa47f25384cdc3"
    },
    {
      "baseSymbol": "WETH",
      "cometAddress": "0xA17581A9E3356d9A858b789D68B4d866e593aE94"
    },
    {
      "baseSymbol": "USDC",
      "cometAddress": "0xF25212E676D1F7F89Cd72fFEe66158f541246445"
    },
    {
      "baseSymbol": "USDbC",
      "cometAddress": "0x9c4ec768c28520B50860ea7a15bd7213a9fF58bf"
    },
    {
      "baseSymbol": "WETH",
      "cometAddress": "0x46e6b214b524310239732D51387075E0e70970bf"
    },
    {
      "baseSymbol": "USDC.e",
      "cometAddress": "0xA5EDBD9646f8dFF606d7448e414884C7d905dCA"
    },
    {
      "baseSymbol": "USDC",
      "cometAddress": "0x9c4ec768c28520B50860ea7a15bd7213a9fF58bf"
    }
  ]
}
```