
eip: <to be assigned>
title: Chain-specific addresses
Author: Lukas, Richard, Pedro, Ligi
discussions-to: <https://ethereum-magicians.org/t/chain-specific-addresses>
status: Draft
type: Standard
category: Interface
created: 2021-08-16

Simple Summary

A standard for displaying [CAIP-10](#) account identifiers in a human readable format

Abstract

This EIP introduced a new address standard to be adapted by wallets and dApps to display chain-specific addresses by mapping human-readable names to [CAIP-3](#) blockchain IDs

Motivation

The need for this EIP emerges from the increasing adoption of non-Ethereum Mainnet chains that use the Ethereum Virtual Machine. In this context, addresses become ambiguous, as the same address may refer to an EOA on chain X or a smart contract on chain Y. This will eventually lead to Ethereum users losing funds due to human error. For example, users sending funds to a smart contract wallet address which was not deployed on a particular chain.

CAIP-10 introduces an account identifier that encodes a CAIP-2 blockchain ID as part of the address. For EVM-based chains, these blockchain IDs are defined in CAIP-3 and leverage [EIP-155](#) chainIDs. However, CAIP-10 targets developers, not end-users. These account identifiers are not meant to be displayed to users in dApps or wallets, and they were optimized for developer interoperability, rather than human readability.

To solve the initial problem of user-facing addresses being ambiguous in a multichain context, we need to extend CAIP-10 with a user-facing format of displaying these account identifiers.

Specification

This EIP extends CAIP-10 with a standard for mapping EVM-based blockchain IDs (CAIP-3) to a human-readable blockchain short name, as defined in [ethereum-lists/chains](https://ethereum-lists.github.io/chains/).

Syntax

A chain-specific address is prefixed with a chain shortName, separated with a colon sign (:).

Chain-specific address = "shortName" ":" "address"

- shortName = STRING
- address = STRING

Semantics

shortName is mandatory and MUST be a valid short name from [ethereum-lists/chains](https://ethereum-lists.github.io/chains)

address is mandatory and MUST be a EIP-55 compatible hexadecimal address

Examples



```
eth:0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
ovm:0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
poly:0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
xdai:0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
arb:0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
bsc:0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
```

Resolution Method

Chain-specific addresses are resolved to CAIP-10 account identifiers using [ethereum-lists/chains](https://ethereum-lists.github.io/chains):

EIP-*tbd* chain-specific address	CAIP-10 account identifier
----------------------------------	----------------------------

eth: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe	eip155:1: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
ovm: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe	eip155:10: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
xdai: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe	eip155:100: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe
poly: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe	eip155:137: 0x0DA0C3e52C977Ed3cBc641fF02DD271c3ED55aFe

Rationale

- CAIP-10 account identifiers are not suitable for user-facing addresses that are chain-specific
- CAIP-10 identifiers are still the preferred option for development purposes, as they are ecosystem-agnostic and work best with chain-splits

Backwards compatibility

tbd